Journal of Computational Physics 319 (2016) 294–323

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A staggered space–time discontinuous Galerkin method for the three-dimensional incompressible Navier–Stokes equations on unstructured tetrahedral meshes

Maurizio Tavelli [a], Michael Dumbser [b]

[a] Department of Mathematics, University of Trento, Via Sommarive 14, I-38050 Trento, Italy
[b] Laboratory of Applied Mathematics, Department of Civil, Environmental and Mechanical Engineering, University of Trento, Via Mesiano 77, I-38123 Trento, Italy

ARTICLE INFO

Article history:
Received 26 January 2016
Received in revised form 2 May 2016
Accepted 3 May 2016
Available online 7 May 2016

Keywords:
High order schemes
Space–time discontinuous Galerkin finite element schemes
Staggered unstructured meshes
Space–time pressure correction algorithm
Incompressible Navier–Stokes equations in 3D

ABSTRACT

In this paper we propose a novel arbitrary high order accurate semi-implicit *space–time* discontinuous Galerkin method for the solution of the three-dimensional incompressible Navier–Stokes equations on *staggered* unstructured curved tetrahedral meshes. As is typical for space–time DG schemes, the discrete solution is represented in terms of space–time basis functions. This allows to achieve very high order of accuracy also in time, which is not easy to obtain for the incompressible Navier–Stokes equations. Similarly to staggered finite difference schemes, in our approach the discrete *pressure* is defined on the *primary* tetrahedral grid, while the discrete *velocity* is defined on a face-based staggered *dual* grid. While staggered meshes are state of the art in classical finite difference schemes for the incompressible Navier–Stokes equations, their use in high order DG schemes is still quite rare. A very simple and efficient Picard iteration is used in order to derive a space–time pressure correction algorithm that achieves also high order of accuracy in time and that avoids the direct solution of global nonlinear systems. Formal substitution of the discrete momentum equation on the dual grid into the discrete continuity equation on the primary grid yields a very sparse five-point block system for the scalar pressure, which is conveniently solved with a matrix-free GMRES algorithm. From numerical experiments we find that the linear system seems to be reasonably well conditioned, since all simulations shown in this paper could be run without the use of any preconditioner, even up to very high polynomial degrees. For a piecewise constant polynomial approximation in time and if pressure boundary conditions are specified at least in one point, the resulting system is, in addition, symmetric and positive definite. This allows us to use even faster iterative solvers, like the conjugate gradient method.

The flexibility and accuracy of high order space–time DG methods on curved unstructured meshes allows to discretize even complex physical domains with very coarse grids in both, space and time. The proposed method is verified for approximation polynomials of degree up to four in space and time by solving a series of typical 3D test problems and by comparing the obtained numerical results with available exact analytical solutions, or with other numerical or experimental reference data.

E-mail addresses: m.tavelli@unitn.it (M. Tavelli), michael.dumbser@unitn.it (M. Dumbser).

http://dx.doi.org/10.1016/j.jcp.2016.05.009

To the knowledge of the authors, this is the first time that a *space–time* discontinuous Galerkin finite element method is presented for the three-dimensional incompressible Navier–Stokes equations on *staggered* unstructured tetrahedral grids.

1. Introduction

The numerical solution of the three dimensional incompressible Navier–Stokes equations represents a very important and challenging research topic, both from a numerical and from an application point of view. In the literature, there are many different approaches that have been proposed for the solution of the incompressible Navier–Stokes equations, for example using classical finite difference methods [1–4] or continuous finite element schemes [5–11]. Very recently, also different high order discontinuous Galerkin (DG) methods have been presented for the solution of the incompressible and the compressible Navier–Stokes equations. The first DG schemes that were able to solve the Navier–Stokes equations were those of Bassi and Rebay [12] and Baumann and Oden [13,14]. Many other methods have been presented in the meantime, see for example [12–27] for a non-exhaustive overview of the ongoing research in this very active field. In most DG schemes, the DG discretization is only used for space discretization, while the time discretization uses standard explicit or implicit time integrators known for ordinary differential equations, following the so-called method of lines approach. The method of lines has also been used by Cockburn and Shu in their well-known series of papers [28–30] on DG schemes for time-dependent nonlinear hyperbolic systems. In contrast to the method of lines approach, the family of space–time discontinuous Galerkin finite element schemes, which was introduced for the first time by van der Vegt et al. in [31–33], treats space and time in a unified manner. This is achieved by using test and basis functions that depend on both space and time, see [34–40] for an overview of recent results. For a very early implementation of continuous space–time finite element schemes, the reader is also referred to [41].

From an application point of view, it is very important to consider the fully three-dimensional Navier–Stokes equations, in order to capture the relevant flow features that are observed in laboratory experiments, see [42–45]. This means that the use of a two-dimensional algorithm is in most cases inappropriate to reproduce the results of physical experiments, even for geometries that can be considered essentially two-dimensional. The importance of fully three-dimensional computations has been shown, for example, in [45–51]. Unfortunately, the mesh generation for complex and realistic 3D geometries is still nowadays quite difficult, and the computational cost of a fully three-dimensional simulation grows very quickly with increasing mesh resolution. In this context, it becomes crucial to use unstructured simplex meshes, since they help to simplify the process of mesh generation significantly compared to unstructured hexahedral meshes. Furthermore, it is at the same time also crucial to use very high order accurate methods in both space and time, since they allow to reduce the total number of elements significantly, compared to low order methods, while keeping at the same time a high level of accuracy of the numerical solution. Since the solution of the incompressible Navier–Stokes equations requires necessarily the solution of large systems of algebraic equations, it is indeed very important to derive a scheme that uses a stencil that is as small as possible, in order to improve the sparsity pattern of the resulting system matrices. It is also desirable to design methods that lead to reasonably well conditioned systems that can be solved with iterative solvers, like the conjugate gradient method [52] or the GMRES algorithm [53].

For structured grids, numerical schemes can be usually derived rather easily in multiple space dimensions, thanks to the particular regularity of the mesh. On the contrary, the development of numerical schemes on general unstructured meshes in three space dimensions is not as straightforward and requires some care in the derivation and the implementation of the method. Particular difficulties of the incompressible Navier–Stokes equations arise from their nonlinearity and from the elliptic nature of the Poisson equation for the pressure, that is also obtained on the discrete level when substituting the momentum equation into the discrete continuity equation. A unified analysis of several variants of the DG method applied to an elliptic model problem has been provided by Arnold et al. in [54].

While the use of *staggered grids* is a very common practice in the finite difference community, its use is not so widespread in the context of high order DG schemes. The first staggered DG schemes, based on a *vertex-based* dual grid, have been proposed in [55,56]. Other recent high order staggered DG schemes that use an *edge-based* dual grid have been forwarded in [57–59]. The advantage in using edge-based staggered grids is that they allow to improve significantly the sparsity pattern of the final linear system that has to be solved for the pressure. Very recently, a new family of *staggered* semi-implicit DG schemes for the solution of the two dimensional shallow water equations was presented by Dumbser & Casulli [59] and Tavelli & Dumbser [60]. Subsequently, these semi-implicit staggered DG schemes have been successfully extended also to the two-dimensional incompressible Navier–Stokes equations by Tavelli & Dumbser in [61,39]. Later, a staggered DG formulation for the 2D incompressible Navier–Stokes equations has been reproposed independently also in [62]. Alternative semi-implicit discontinuous Galerkin schemes on *collocated grids* have been presented, for example, in [63–67]. These semi-implicit schemes try to combine the simplicity of explicit methods for nonlinear PDE with the stability and efficiency of implicit time discretizations.

In this paper we propose a new, arbitrary high order accurate *staggered* space–time discontinuous Galerkin finite element method for the solution of the three-dimensional incompressible Navier–Stokes equations on *curved* unstructured tetrahe-

dral meshes, following some of the ideas outlined in [39] for the two-dimensional case. For that purpose we mimic the philosophy of staggered semi-implicit finite difference schemes, such as discussed and analyzed in [1–4,68–77], where the discrete pressure field is defined on the primary grid, while the discrete velocity field is defined on an edge-based staggered dual grid.

For the staggered space–time DG scheme proposed in this paper, we use a *primal mesh* composed of (curved) tetrahedral elements, and a face-based staggered *dual mesh* that consists of **non-standard five-point** hexahedral elements that are obtained by connecting the three nodes of a face of the primal mesh with the barycenters of the two tetrahedra that share the common face. The face-based dual grid used here corresponds to the choice made also in [78–80,58]. These spatial elements are then extended to space–time control volumes using a simple tensor product in the time direction.

Since all quantities are readily defined where they are needed, our staggered DG scheme does *not* require the use of Riemann solvers (numerical flux functions), apart from the nonlinear convective terms, which are treated in a conventional way. Note that this special feature is *not* standard for DG schemes, which typically require numerical fluxes or penalty terms due to the presence of jumps of the discrete solution at the element boundaries, in particular for the discretization of second and higher order derivatives, see [81–83,17]. For the nonlinear convective part of the incompressible Navier–Stokes equations, we use a standard DG scheme for hyperbolic PDE on the main grid, based on the local Lax–Friedrichs (Rusanov) flux [84]. For that purpose, the velocity field is first interpolated from the dual grid to the main grid, as suggested in [59]. This allows us to use the same staggered space–time DG scheme *again* to discretize also the viscous terms, where now the *velocity gradient* that is needed for the evaluation of the viscous fluxes is computed on the face-based *staggered* dual grid. In this way, we can avoid again the use of numerical flux functions for the viscous terms, and furthermore, the structure of the resulting linear systems for the viscous terms is very similar to the pressure system. In some sense, our new discretization of the viscous terms can be interpreted as a *lifting operator* in the sense of Bassi and Rebay [12], but producing discrete gradients on the *staggered* dual grid.

The discrete momentum equation is then inserted into the discrete continuity equation in order to obtain the discrete form of the pressure Poisson equation. Once the new pressure field is known, the velocity vector field can subsequently be updated directly. A very simple Picard iteration that embraces the entire space–time DG scheme in each time step is used in order to achieve arbitrary high order of accuracy in time also for the nonlinear convective and viscous terms, without introducing a nonlinearity in the system for the pressure.

In order to compare the staggered DG algorithm with the same DG scheme on a collocated grid, let us consider in the following only the case of first order in time and high order in space and the coupling of the pressure gradient in the momentum equation to the divergence constraint of the velocity, neglecting for a moment the presence of the nonlinear convective and the viscous terms:

Thanks to the use of a *staggered grid*, our discretization leads to a very sparse five-point block system, with the scalar pressure as the *only* unknown quantity.[1] Note that the *same algorithm* on a *collocated grid* would produce a **17-point** stencil, since it would also involve neighbors of neighbors.[2] On the other hand, if one does not substitute the momentum equation into the continuity equation on a collocated grid, one could still obtain a five point stencil, but with the pressure *and* the velocity vector as unknowns (leading to a saddle point problem), hence the final system to solve is four times larger than the corresponding system of our staggered DG scheme. It is therefore very clear that even in the DG context, the use of a staggered mesh is very beneficial, since it allows to produce a linear system with the smallest possible stencil and with the smallest number of unknowns, compared to similar approaches on a collocated mesh.

One of the key novelties in this paper w.r.t. [39] is indeed the novel discretization of the viscous terms mentioned above, which makes again efficient use of the combination of primal and staggered dual grid, while in [39] a penalty approach based on the ideas of Gassner et al. [17] was employed directly on the dual grid. Another important change introduced in this paper concerns the choice of the basis functions on the staggered dual grid, due to the appearing non-standard five-point hexahedra. While in the 2D case described in [39] the dual grid consisted of simple quadrilateral elements, and thus a natural nodal basis was available, the straightforward 3D extension of the nodal basis used in [39] can encounter singularities, hence requiring either the choice of a more sophisticated non-polynomial nodal basis, or the use of a *modal basis*, as used in this paper.

The rest of the paper is organized as follows: in Section 2 we derive and present the new numerical method. Section 2.5 contains the details about the discretization of the nonlinear convective terms on the main grid, while the velocity gradients for the viscous terms are discretized again on the face-based staggered dual mesh. In Section 2.7 we discuss the important

[1] Note that with *scalar pressure* we mean all degrees of freedom that define the discrete pressure in the DG scheme. As a consequence, all operations performed on an element potentially involve all degrees of freedom of that element, and for that reason the final pressure system is a *block* five-point system, where the *blocks* refer to the operations on the DOF inside each element.

[2] The discrete continuity equation of a DG scheme on a collocated grid involves the velocity in the element itself and in its four neighbor elements, due to the numerical flux involving the jumps on the element boundaries. Furthermore, in the discrete momentum equation the velocity field in each tetrahedral element depends on the pressure in the cell itself and in its four neighbors. Inserting now the momentum equation into the continuity equation on the discrete level involves a total of $1+4+4\cdot 3=17$ elements for the pressure!

On a staggered mesh instead, the discrete continuity equation involves only the velocities of the four dual elements associated with the faces of the primary element. The discrete momentum equation written on the face-based dual grid only involves the pressure of the *two* tetrahedra that share the common face. Hence, substituting the momentum equation into the continuity equation leads to a $1+4=5$ point stencil for the pressure, which involves only the element and its four neighbors.

special case of a high order DG discretization in space, while using only a piecewise constant polynomial approximation in time, leading to symmetric positive definite systems for the pressure and the viscous terms. Finally, in Section 3 the new numerical scheme proposed in this paper is run on a set of 3D benchmark problems, comparing the numerical results either with existing analytical or numerical reference solutions, or with available experimental results. The paper closes with some concluding remarks provided in Section 4.

2. Staggered space–time DG scheme for the 3D incompressible Navier–Stokes equations

2.1. Governing equations

The three-dimensional incompressible Navier–Stokes equations can be written as

$$\frac{\partial \mathbf{v}}{\partial t} + \nabla \cdot \mathbf{F}_c + \nabla p = \nabla \cdot (\nu \nabla \mathbf{v}) + \mathbf{S}, \tag{1}$$

$$\nabla \cdot \mathbf{v} = 0, \tag{2}$$

where $\mathbf{x} = (x, y, z)$ is the vector of spatial coordinates and t denotes the time; $p = P/\rho$ indicates the normalized fluid pressure; P is the physical pressure and ρ is the constant fluid density; $\nu = \mu/\rho$ is the kinematic viscosity coefficient; $\mathbf{v} = (u, v, w)$ is the velocity vector; u, v and w are the velocity components in the x, y and z direction, respectively; $\mathbf{S} = \mathbf{S}(\mathbf{x}, t)$ is a vector of given source terms; $\mathbf{F}_c = \mathbf{v} \otimes \mathbf{v}$ is the flux tensor of the nonlinear convective terms, namely:

$$\mathbf{F}_c = \begin{pmatrix} uu & uv & uw \\ vu & vv & vw \\ wu & wv & ww \end{pmatrix}.$$

The viscosity term can be grouped with the nonlinear convective term, i.e. the momentum Eq. (1) then reads

$$\frac{\partial \mathbf{v}}{\partial t} + \nabla \cdot \mathbf{F} + \nabla p = \mathbf{S}, \tag{3}$$

where $\mathbf{F} = \mathbf{F}(\mathbf{v}, \nabla \mathbf{v}) = \mathbf{F}_c(\mathbf{v}) - \nu \nabla \mathbf{v}$ is the nonlinear flux tensor that depends on the velocity and its gradient.

2.2. Staggered unstructured mesh and associated space–time basis functions

Throughout this paper we use a main grid that is composed of (eventually curved) tetrahedral simplex elements, and a staggered face-based dual grid, consisting in non-standard five-point hexahedral elements. These spatial control volumes are then extended to space–time control volumes using a tensor product in time direction. In the following, the staggered mesh in space is described in detail and is subsequently also extended to the time direction. The main notation is taken as the one presented for the two dimensional method proposed in [39] and is summarized here for the three dimensional case.

2.2.1. Staggered space–time control volumes

The spatial computational domain Ω is covered with a set of N_e non-overlapping tetrahedral elements $\boldsymbol{T}_i$ with $i = 1 \dots N_e$. By denoting with N_d the total number of faces, the j-th face will be called Γ_j. $\mathcal{B}(\Omega)$ denotes the set of indices j corresponding to boundary faces. The indices of the four faces of each tetrahedron $\boldsymbol{T}_i$ constitute the set S_i defined by $S_i = \{j \in [1, N_d] \mid \Gamma_j \text{ is a face of } \boldsymbol{T}_i\}$. For every $j \in [1 \dots N_d] - \mathcal{B}(\Omega)$ there exist two tetrahedra that share a common face Γ_j. We assign arbitrarily a left and a right element, called $\boldsymbol{T}_{\ell(j)}$ and $\boldsymbol{T}_{r(j)}$, respectively. The standard positive direction is assumed to be from left to right. Let $\vec{n}_j$ denote the unit normal vector defined on the face number j and that is oriented with respect to the positive direction from left to right. For every tetrahedral element number i and face number $j \in S_i$, the index of the neighbor tetrahedron that shares the common face Γ_j is denoted by $\wp(i, j)$.

For every $j \in [1, N_d] - \mathcal{B}(\Omega)$ the dual element (a non-standard 5-point hexahedron) associated with Γ_j is called $\boldsymbol{H}_j$ and it is defined by the two centers of gravity of $\boldsymbol{T}_{\ell(j)}$ and $\boldsymbol{T}_{r(j)}$ and the three vertices of Γ_j, see also [78,80,60]. We denote by $\boldsymbol{T}_{i,j} = \boldsymbol{H}_j \cap \boldsymbol{T}_i$ the intersection element for every i and $j \in S_i$. Figs. 1 and 2 summarize the notation used on the main tetrahedral mesh and on the associated dual grid. We extend our definitions on the main grid to the dual one, namely: N_l is the total amount of sides of $\boldsymbol{H}_j$; Γ_l indicates the l-th side; $\forall j$, the set of sides l of j is indicated with S_j; $\forall l$, $\ell_{jl}(l)$ and $r_{jl}(l)$ are the left and the right hexahedral element, respectively; $\vec{n}_l$ is the standard normal vector defined on l and assumed positive with respect to the standard orientation on l (defined, as for the main grid, from the left to the right).

In the time direction we cover the time interval $[0, T]$ with a sequence of times $0 = t^0 < t^1 < t^2 \dots < t^N < t^{N+1} = T$. We denote the time step between t^n and t^{n+1} by $\Delta t^{n+1} = t^{n+1} - t^n$ and the associated time interval by $T^{n+1} = [t^n, t^{n+1}]$, for $n = 0 \dots N$. In order to ease the notation, sometimes we will simply write $\Delta t = \Delta t^{n+1}$. In this way the generic space–time element defined in the time interval $[t^n, t^{n+1}]$ is given by $\boldsymbol{T}_i^{st} = \boldsymbol{T}_i \times T^{n+1}$ for the main grid and $\boldsymbol{H}_j^{st} = \boldsymbol{H}_j \times T^{n+1}$ for the dual grid.

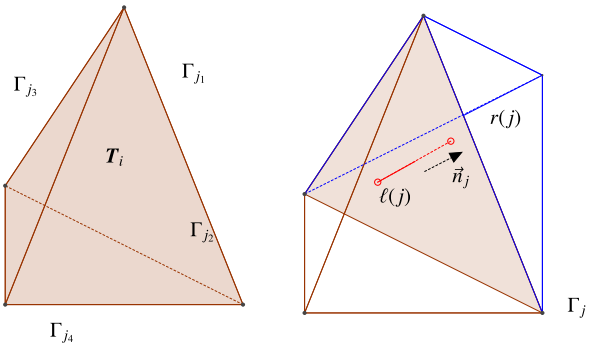

Fig. 1. A tetrahedral element of the primary mesh with $S_i = \{j_1, j_2, j_3, j_4\}$ (left) and the standard orientation used throughout this paper (right).

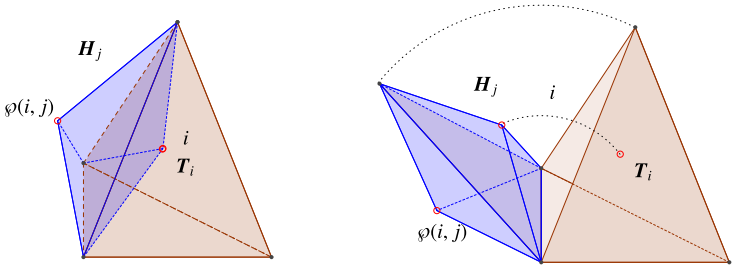

Fig. 2. An example of a dual element (a non-standard 5-point hexahedron, highlighted in blue) associated with the face Γ_j. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

2.3. Space–time basis functions

We first construct the spatial basis functions and then we extend them to the time direction using a simple tensor product. For tetrahedral elements, the basis functions are generated on a standard reference tetrahedron, defined by $T_{ref} = \{(\xi, \eta, \zeta) \in \mathbb{R}^3 \mid 0 \le \xi + \eta + \zeta \le 1\}$. We write the basis function on the reference element as

$$\phi_k(\xi, \eta, \zeta) = \sum_{r_1=0}^{p} \sum_{r_2=0}^{p-r_1} \sum_{r_3=0}^{p-r_2-r_1} \alpha_{kr}\, \xi^{r_1} \eta^{r_2} \zeta^{r_3} := \alpha_{kr} \boldsymbol{\xi}^r, \tag{4}$$

for some coefficients α_{kr} and the multi-index $r = (r_1, r_2, r_3)$. We then define $N_\phi = \frac{(p+1)(p+2)(p+3)}{6}$ nodal points $\boldsymbol{\xi}_j = (\xi_{j_1}, \eta_{j_2}, \zeta_{j_3}) = (j_1/p, j_2/p, j_3/p)$, with the multi-index $j = (j_1, j_2, j_3)$ and $0 \le j_1 + j_2 + j_3 \le p$, as in standard conforming finite elements. We then impose the classical interpolation condition for nodal finite elements $\phi_k(\boldsymbol{\xi}_j) = \delta_{kj}$, with the usual Kronecker symbol δ_{kj}. This means that we have chosen a nodal basis which is defined by the Lagrange interpolation polynomials that pass through the nodes given by the standard nodes of conforming finite elements. This leads to the linear system $\alpha_{kr} \boldsymbol{\xi}_j^r = \delta_{kj}$ for the coefficients α_{kr} that can be solved analytically for every polynomial degree p on the reference tetrahedron. In this way we obtain N_ϕ basis functions on T_{ref}, $\{\phi_k\}_{k \in [1, N_\phi]}$. The connection between the reference coordinates $\boldsymbol{\xi}$ and the physical coordinates $\mathbf{x}$ is performed by the map $T(\cdot, \boldsymbol{T}_i) = T_i : \boldsymbol{T}_i \longrightarrow T_{ref}$ for every $i = 1 \dots N_e$ and its inverse, called $T^{-1}(\cdot, \boldsymbol{T}_i) = T_i^{-1} : \boldsymbol{T}_i \longleftarrow T_{ref}$. The maps from the physical to the reference coordinates can be constructed following a classical sub-parametric or a complete iso-parametric approach and in general we will write, for all $i = 1 \dots N_e$, $\phi_k^{(i)}(x, y, z) = \phi_k(T_i(x, y, z))$.

Unfortunately, it is not so easy to construct a similar nodal basis on the dual mesh, due to the use of non-standard 5-point hexahedral elements. As discussed in [85], the definition of basis functions based on Lagrange interpolation polynomials on this kind of element is problematic, since for special configurations of the vertex coordinates of the dual elements, the linear system to be solved for the classical interpolation condition of a nodal basis can become singular. This does not allow the construction of a nodal polynomial basis for a generic element $\boldsymbol{H}_j$ and therefore one has to pass to rational functions of polynomials instead of using simple polynomial functions in that case.

Therefore, for the basis functions on the dual grid directly we choose a simple Taylor-type modal basis [86] directly in the physical space, hence the basis functions will consequently depend on the element $j \in [1, N_d]$. The basis functions read

$$\psi_k^{(j)}(\mathbf{x}) = \frac{\left(x - x_0^{(j)}\right)^{k_1} \left(y - y_0^{(j)}\right)^{k_2} \left(z - z_0^{(j)}\right)^{k_3}}{h_j^{k_1+k_2+k_3}}, \tag{5}$$

where $\mathbf{x}_0^j = (x, y, z)_0^{(j)}$ is the center of the dual element and h_j is a characteristic length of $\boldsymbol{H}_j$ used for scaling the basis. Here, $0 \leq k_1 + k_2 + k_3 \leq p$, i.e. we use the optimal number of polynomials of degree p in three space dimensions, namely $N_\psi = N_\phi$. With this choice we get only a modal basis for the dual hexahedral elements, i.e. if the convective term is directly computed on the dual mesh according to the natural extension of the method proposed in [39], then it has to be computed according to a modal approach, which is more expensive than a nodal one.

Finally, the time basis functions are constructed on a reference interval $I = [0, 1]$ for polynomials of degree p_γ. In this case the resulting $N_\gamma = p_\gamma + 1$ basis functions $\{\gamma_k\}_{k\in[1,N_\gamma]}$ are defined as the Lagrange interpolation polynomials passing through the Gauss–Legendre quadrature points for the unit interval. For every time interval $[t^n, t^{n+1}]$, the map between the reference interval and the physical one is simply given by $t = t^n + \tau\,\Delta t^{n+1}$ for every $\tau \in [0, 1]$. Using the tensor product we can finally construct the basis functions on the space–time elements $\boldsymbol{T}_i^{st}$ and $\boldsymbol{H}_j^{st}$ such as $\tilde{\phi}(\xi, \eta, \zeta, \tau) = \phi(\xi, \eta, \zeta) \cdot \gamma(\tau)$ and $\tilde{\psi}^{(j)}(x, y, z, t) = \psi^{(j)}(x, y, z) \cdot \gamma(\tau(t))$. The total number of basis functions becomes $N_\phi^{st} = N_\phi \cdot N_\gamma$ and $N_\psi^{st} = N_\psi \cdot N_\gamma$.

2.4. Staggered semi-implicit space–time DG scheme

The discrete pressure p_h is defined on the main grid, namely $p_h(\mathbf{x}, t)|_{\boldsymbol{T}_i^{st}} = p_i(\mathbf{x}, t) = p_i$, while the discrete velocity vector field $\mathbf{v}_h$ is defined on the dual grid, namely $\mathbf{v}_h(\mathbf{x}, t)|_{\boldsymbol{H}_j^{st}} = \mathbf{v}_j(\mathbf{x}, t) = \mathbf{v}_j$.

The numerical solution of (2)–(3) is represented in each space–time element $\boldsymbol{T}_i^{st}$ and $\boldsymbol{H}_j^{st}$ between times t^n and t^{n+1} by piecewise polynomials as

$$p_i(\mathbf{x}, t) = \sum_{l=1}^{N_\phi^{st}} \tilde{\phi}_l^{(i)}(\mathbf{x}, t)\,\hat{p}_{l,i}^{n+1} =: \tilde{\boldsymbol{\phi}}^{(i)}(\mathbf{x}, t)\,\hat{\mathbf{p}}_i^{n+1}, \tag{6}$$

$$\mathbf{v}_j(\mathbf{x}, t) = \sum_{l=1}^{N_\psi^{st}} \tilde{\psi}_l^{(j)}(\mathbf{x}, t)\,\hat{\mathbf{v}}_{l,j}^{n+1} =: \tilde{\boldsymbol{\psi}}^{(j)}(\mathbf{x}, t)\,\hat{\mathbf{v}}_j^{n+1}, \tag{7}$$

where the vector of basis functions $\tilde{\boldsymbol{\phi}}(\mathbf{x}, t)$ is generated from $\tilde{\boldsymbol{\phi}}(\xi, \eta, \zeta, \tau)$ on $T_{std} \times [0, 1]$ while $\tilde{\boldsymbol{\psi}}^{(j)}(\mathbf{x}, t)$ is defined for every $j \in [1 \ldots N_d] - \mathcal{B}(\Omega)$ directly in the physical space.

A weak formulation of the continuity equation (2) is obtained by multiplying it with a test function $\tilde{\phi}_k^{(i)}$ and integrating over the space–time control volume $\boldsymbol{T}_i^{st}$, for every $k = 1 \ldots N_\phi^{st}$. The resulting weak formulation reads

$$\int_{\boldsymbol{T}_i^{st}} \tilde{\phi}_k^{(i)} \nabla \cdot \mathbf{v}\, d\mathbf{x}dt = 0, \tag{8}$$

with $d\mathbf{x} = dxdydz$. Similarly, multiplication of the momentum equation (3) by the test function $\tilde{\psi}_k^{(j)}$ and integrating over a control volume $\boldsymbol{H}_j^{st}$ yields

$$\int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \left(\frac{\partial \mathbf{v}}{\partial t} + \nabla \cdot \mathbf{F} \right) d\mathbf{x}dt + \int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \nabla p\; d\mathbf{x}dt = \int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \mathbf{S}\, d\mathbf{x}dt, \tag{9}$$

for every $j = 1 \ldots N_d$ and $k = 1 \ldots N_\psi^{st}$. Using integration by parts Eq. (8) reads

$$\oint_{\partial \boldsymbol{T}_i^{st}} \tilde{\phi}_k^{(i)} \mathbf{v} \cdot \vec{n}_i\, dSdt - \int_{\boldsymbol{T}_i^{st}} \nabla \tilde{\phi}_k^{(i)} \cdot \mathbf{v}\, d\mathbf{x}dt = 0, \tag{10}$$

where $\vec{n}_i$ indicates the outward pointing unit normal vector. Due to the discontinuity of p_h and $\mathbf{v}_h$, equations (9) and (10) have to be split as follows:

$$\sum_{j \in S_i} \left(\int_{\Gamma_j^{st}} \tilde{\phi}_k^{(i)} \mathbf{v}_j \cdot \vec{n}_{ij}\, dSdt - \int_{\boldsymbol{T}_{i,j}^{st}} \nabla \tilde{\phi}_k^{(i)} \cdot \mathbf{v}_j\, d\mathbf{x}dt \right) = 0, \tag{11}$$

and

$$\int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \left(\frac{\partial \mathbf{v}_j}{\partial t} + \nabla \cdot \mathbf{F} \right) d\mathbf{x}dt + \int_{\boldsymbol{T}_{\ell(j),j}^{st}} \tilde{\psi}_k^{(j)} \nabla p_{\ell(j)}\, d\mathbf{x}dt + \int_{\boldsymbol{T}_{r(j),j}^{st}} \tilde{\psi}_k^{(j)} \nabla p_{r(j)}\, d\mathbf{x}dt + \int_{\Gamma_j^{st}} \tilde{\psi}_k^{(j)} \left(p_{r(j)} - p_{\ell(j)} \right) \vec{n}_j\, dSdt$$
$$= \int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \mathbf{S}\, d\mathbf{x}dt, \tag{12}$$

where $\vec{n}_{ij} = \vec{n}_i|_{\Gamma_j^{st}}$; $\boldsymbol{T}_{i,j}^{st} = \boldsymbol{T}_{i,j} \times T^{n+1}$; and $\Gamma_j^{st} = \Gamma_j \times T^{n+1}$. Note that the pressure has a discontinuity along Γ_j^{st} inside the hexahedral element $\boldsymbol{H}_j^{st}$ and hence the pressure gradient in (9) needs to be interpreted in the sense of distributions, as in path-conservative finite volume schemes [87,88]. This leads to the jump terms present in (12), see [39]. Alternatively, the same jump term can be produced also via forward and backward integration by parts, see e.g. the well-known work of Bassi and Rebay [12]. Using definitions (6) and (7), we rewrite the above equations as

$$\sum_{j \in S_i} \left(\int_{\Gamma_j^{st}} \tilde{\phi}_k^{(i)} \tilde{\psi}_l^{(j)} \vec{n}_{ij} dSdt \cdot \hat{\mathbf{v}}_{l,j}^{n+1} - \int_{\boldsymbol{T}_{i,j}^{st}} \nabla \tilde{\phi}_k^{(i)} \tilde{\psi}_l^{(j)}\, d\mathbf{x}dt \cdot \hat{\mathbf{v}}_{l,j}^{n+1} \right) = 0, \tag{13}$$

and

$$\int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \frac{\partial \mathbf{v}_j}{\partial t}\, d\mathbf{x}dt + \int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \nabla \cdot \mathbf{F}\, d\mathbf{x}dt + \int_{\boldsymbol{T}_{\ell(j),j}^{st}} \tilde{\psi}_k^{(j)} \nabla \tilde{\phi}_l^{(\ell(j))}\, d\mathbf{x}dt\ \hat{p}_{l,\ell(j)}^{n+1} + \int_{\boldsymbol{T}_{r(j),j}^{st}} \tilde{\psi}_k^{(j)} \nabla \tilde{\phi}_l^{(r(j))}\, d\mathbf{x}dt\ \hat{p}_{l,r(j)}^{n+1}$$
$$+ \int_{\Gamma_j^{st}} \tilde{\psi}_k^{(j)} \tilde{\phi}_l^{(r(j))} \vec{n}_j dSdt\, \hat{p}_{l,r(j)}^{n+1} - \int_{\Gamma_j^{st}} \tilde{\psi}_k^{(j)} \tilde{\phi}_l^{(\ell(j))} \vec{n}_j dSdt\, \hat{p}_{l,\ell(j)}^{n+1} = \int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \mathbf{S}\, d\mathbf{x}dt, \tag{14}$$

where we have used the standard summation convention for the repeated index l. Integrating the first integral in (14) by parts in time we obtain

$$\int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \frac{\partial \mathbf{v}_j}{\partial t}\, d\mathbf{x}dt = \int_{\boldsymbol{H}_j} \tilde{\psi}_k^{(j)}(\mathbf{x}, t^{n+1}) \mathbf{v}_j(\mathbf{x}, t^{n+1})\, d\mathbf{x} - \int_{\boldsymbol{H}_j} \tilde{\psi}_k^{(j)}(\mathbf{x}, t^{n}) \mathbf{v}_j(\mathbf{x}, t^{n})\, d\mathbf{x} - \int_{\boldsymbol{H}_j^{st}} \frac{\partial \tilde{\psi}_k^{(j)}}{\partial t} \mathbf{v}_j(\mathbf{x}, t)\, d\mathbf{x}dt. \tag{15}$$

In Eq. (15) we can recognize the fluxes between the current space–time element $\boldsymbol{H}_j \times T^{n+1}$, the future and the past space–time elements, as well as an internal contribution that connects in an asymmetric way the degrees of freedom inside the element $\boldsymbol{H}_j^{st}$. Note that the asymmetry appears only in the volume contribution in (15). For the spatial integral at time t^n we will insert the boundary-extrapolated numerical solution from the previous time step, which corresponds to upwinding in time direction due to the causality principle. By substituting Eq. (15) into (14) and using the causality principle, we obtain the following weak formulation of the momentum equation:

$$\left(\int_{\boldsymbol{H}_j} \tilde{\psi}_k^{(j)}(\mathbf{x}, t^{n+1}) \tilde{\psi}_l^{(j)}(\mathbf{x}, t^{n+1})\, d\mathbf{x} - \int_{\boldsymbol{H}_j^{st}} \frac{\partial \tilde{\psi}_k^{(j)}}{\partial t} \tilde{\psi}_l^{(j)}\, d\mathbf{x}dt \right) \hat{\mathbf{v}}_{l,j}^{n+1} - \int_{\boldsymbol{H}_j} \tilde{\psi}_k^{(j)}(\mathbf{x}, t^{n}) \tilde{\psi}_l^{(j)}(\mathbf{x}, t^{n})\, d\mathbf{x}\, \hat{\mathbf{v}}_{l,j}^{n} + \int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \nabla \cdot \mathbf{F}\, d\mathbf{x}$$
$$+ \int_{\boldsymbol{T}_{\ell(j),j}^{st}} \tilde{\psi}_k^{(j)} \nabla \tilde{\phi}_l^{(\ell(j))}\, d\mathbf{x}\, \hat{p}_{l,\ell(j)}^{n+1} + \int_{\boldsymbol{T}_{r(j),j}^{st}} \tilde{\psi}_k^{(j)} \nabla \tilde{\phi}_l^{(r(j))}\, d\mathbf{x}\, \hat{p}_{l,r(j)}^{n+1} + \int_{\Gamma_j^{st}} \tilde{\psi}_k^{(j)} \tilde{\phi}_l^{(r(j))} \vec{n}_j dS\, \hat{p}_{l,r(j)}^{n+1}$$
$$- \int_{\Gamma_j^{st}} \tilde{\psi}_k^{(j)} \tilde{\phi}_l^{(\ell(j))} \vec{n}_j dS\, \hat{p}_{l,\ell(j)}^{n+1}$$
$$= \int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)} \mathbf{S}\, d\mathbf{x}dt. \tag{16}$$

For every i and j, Eqs. (13) and (16) can be written in a compact matrix form as

$$\sum_{j \in S_i} \mathcal{D}_{i,j} \hat{\mathbf{v}}_j^{n+1} = 0, \tag{17}$$

and

$$\left(\boldsymbol{M}_j^{+} - \boldsymbol{M}_j^{\circ}\right)\hat{\mathbf{v}}_j^{n+1} - \boldsymbol{M}_j^{-}\hat{\mathbf{v}}_j^{n} + \Upsilon_j(\mathbf{v}, \nabla\mathbf{v}) + \boldsymbol{\mathcal{R}}_j\hat{\mathbf{p}}_{r(j)}^{n+1} - \boldsymbol{\mathcal{L}}_j\hat{\mathbf{p}}_{\ell(j)}^{n+1} = \boldsymbol{\mathcal{S}}_j, \tag{18}$$

respectively, where:

$$\boldsymbol{M}_j^{+} = \int_{\boldsymbol{H}_j} \tilde{\psi}_k^{(j)}(\mathbf{x}, t(1))\tilde{\psi}_l^{(j)}(\mathbf{x}, t(1))\, d\mathbf{x}, \tag{19}$$

$$\boldsymbol{M}_j^{-} = \int_{\boldsymbol{H}_j} \tilde{\psi}_k^{(j)}(\mathbf{x}, t(0))\tilde{\psi}_l^{(j)}(\mathbf{x}, t(1))\, d\mathbf{x}, \tag{20}$$

$$\boldsymbol{M}_j^{\circ} = \int_{\boldsymbol{H}_j^{st}} \frac{\partial\tilde{\psi}_k^{(j)}}{\partial t}\tilde{\psi}_l^{(j)}\, d\mathbf{x}dt, \tag{21}$$

$$\Upsilon_j = \int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)}\nabla\cdot\mathbf{F}\, d\mathbf{x}dt \tag{22}$$

$$\boldsymbol{\mathcal{D}}_{i,j} = \int_{\Gamma_j^{st}} \tilde{\phi}_k^{(i)}\tilde{\psi}_l^{(j)}\vec{n}_{ij}dSdt - \int_{\boldsymbol{T}_{i,j}^{st}} \nabla\tilde{\phi}_k^{(i)}\tilde{\psi}_l^{(j)}\, d\mathbf{x}dt, \tag{23}$$

$$\boldsymbol{\mathcal{R}}_j = \int_{\Gamma_j^{st}} \tilde{\psi}_k^{(j)}\tilde{\phi}_l^{(r(j))}\vec{n}_j dSdt + \int_{\boldsymbol{T}_{r(j),j}^{st}} \tilde{\psi}_k^{(j)}\nabla\tilde{\phi}_l^{(r(j))}\, d\mathbf{x}dt, \tag{24}$$

$$\boldsymbol{\mathcal{L}}_j = \int_{\Gamma_j^{st}} \tilde{\psi}_k^{(j)}\tilde{\phi}_l^{(\ell(j))}\vec{n}_j dSdt - \int_{\boldsymbol{T}_{\ell(j),j}^{st}} \tilde{\psi}_k^{(j)}\nabla\tilde{\phi}_l^{(\ell(j))}\, d\mathbf{x}dt, \tag{25}$$

$$\boldsymbol{\mathcal{S}}_j = \int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)}\mathbf{S}\, d\mathbf{x}dt. \tag{26}$$

Note how $\boldsymbol{M}_j^{\circ}$ introduces, for $p_\gamma > 0$, an asymmetric contribution that will lead to an asymmetry of the main system for the discrete pressure. The action of matrices $\boldsymbol{\mathcal{L}}$ and $\boldsymbol{\mathcal{R}}$ can be generalized by introducing the new matrix $\boldsymbol{\mathcal{Q}}_{i,j}$, defined as

$$\boldsymbol{\mathcal{Q}}_{i,j} = \int_{\boldsymbol{T}_{i,j}^{st}} \tilde{\psi}_k^{(j)}\nabla\tilde{\phi}_l^{(i)}\, d\mathbf{x}dt - \int_{\Gamma_j^{st}} \tilde{\psi}_k^{(j)}\tilde{\phi}_l^{(i)}\sigma_{i,j}\vec{n}_j dsdt, \tag{27}$$

where $\sigma_{i,j}$ is a sign function defined by

$$\sigma_{i,j} = \frac{r(j) - 2i + \ell(j)}{r(j) - \ell(j)}. \tag{28}$$

In this way $\boldsymbol{\mathcal{Q}}_{\ell(j),j} = -\boldsymbol{\mathcal{L}}_j$ and $\boldsymbol{\mathcal{Q}}_{r(j),j} = \boldsymbol{\mathcal{R}}_j$, and then Eq. (18) becomes in terms of $\boldsymbol{\mathcal{Q}}$

$$\left(\boldsymbol{M}_j^{+} - \boldsymbol{M}_j^{\circ}\right)\hat{\mathbf{v}}_j^{n+1} - \boldsymbol{M}_j^{-}\hat{\mathbf{v}}_j^{n} + \Upsilon_j(\mathbf{v}, \nabla\mathbf{v}) + \boldsymbol{\mathcal{Q}}_{r(j),j}\hat{\mathbf{p}}_{r(j)}^{n+1} + \boldsymbol{\mathcal{Q}}_{\ell(j),j}\hat{\mathbf{p}}_{\ell(j)}^{n+1} = \boldsymbol{\mathcal{S}}_j, \tag{29}$$

or, equivalently,

$$\left(\boldsymbol{M}_j^{+} - \boldsymbol{M}_j^{\circ}\right)\hat{\mathbf{v}}_j^{n+1} - \boldsymbol{M}_j^{-}\hat{\mathbf{v}}_j^{n} + \Upsilon_j(\mathbf{v}, \nabla\mathbf{v}) + \boldsymbol{\mathcal{Q}}_{i,j}\hat{\mathbf{p}}_i^{n+1} + \boldsymbol{\mathcal{Q}}_{\wp(i,j),j}\hat{\mathbf{p}}_{\wp(i,j)}^{n+1} = \boldsymbol{\mathcal{S}}_j. \tag{30}$$

In order to ease the notation we will use $\boldsymbol{M}_j = \boldsymbol{M}_j^{+} - \boldsymbol{M}_j^{\circ}$. Hence, the discrete equations (17)–(18) read as follows (saddle point problem):

$$\sum_{j\in S_i} \boldsymbol{\mathcal{D}}_{i,j}\hat{\mathbf{v}}_j^{n+1} = 0, \tag{31}$$

$$\boldsymbol{M}_j\hat{\mathbf{v}}_j^{n+1} - \boldsymbol{M}_j\widehat{\boldsymbol{F}\mathbf{v}}_j + \boldsymbol{\mathcal{Q}}_{r(j),j}\hat{\mathbf{p}}_{r(j)}^{n+1} + \boldsymbol{\mathcal{Q}}_{\ell(j),j}\hat{\mathbf{p}}_{\ell(j)}^{n+1} = 0, \tag{32}$$

where $\widehat{\boldsymbol{F}\mathbf{v}}_j$ is an appropriate discretization of the nonlinear convective, viscous and source terms that will be presented later. Formal substitution of the discrete velocity field given by the momentum equation (32) into the discrete continuity equation (31), see also [69,59], yields a discrete pressure equation

$$\sum_{j\in S_i} \mathcal{D}_{i,j}\boldsymbol{M}_j^{-1}\mathcal{Q}_{i,j}\hat{\boldsymbol{p}}_i^{n+1} + \sum_{j\in S_i} \mathcal{D}_{i,j}\boldsymbol{M}_j^{-1}\mathcal{Q}_{\wp(i,j),j}\hat{\boldsymbol{p}}_{\wp(i,j)}^{n+1} = \sum_{j\in S_i} \mathcal{D}_{i,j}\widehat{\boldsymbol{F}\mathbf{v}}_j. \tag{33}$$

Eq. (33) above represents a block five-point system for the pressure degrees of freedom $\hat{\boldsymbol{p}}_i^{n+1}$ inside each element and where the *blocks* are due to the action of the matrices (19)–(25) on the degrees of freedom. The saddle point problem (31) & (32) and the pressure equation (33) are completely equivalent at the discrete level, since the latter has been obtained from the former only via direct substitution of (32) into (31).

2.5. Nonlinear convective and viscous terms

We now have to choose a proper discretization for the nonlinear convective and viscous terms. As discussed in [39] we introduce a simple Picard iteration to update the information about the pressure, but without introducing any nonlinearity into the final system for the pressure. Hence, for $k = 1, N_{pic}$, we rewrite system (33) as

$$\sum_{j\in S_i} \mathcal{D}_{i,j}\boldsymbol{M}_j^{-1}\mathcal{Q}_{i,j}\hat{\boldsymbol{p}}_i^{n+1,k+1} + \sum_{j\in S_i} \mathcal{D}_{i,j}\boldsymbol{M}_j^{-1}\mathcal{Q}_{\wp(i,j),j}\hat{\boldsymbol{p}}_{\wp(i,j)}^{n+1,k+1} = \sum_{j\in S_i} \mathcal{D}_{i,j}\widehat{\boldsymbol{F}\mathbf{v}}_j^{n+1,k+\frac{1}{2}}. \tag{34}$$

The right side of Eq. (34) can be computed by using the velocity field at the Picard iteration k and including the viscous effect implicitly, using a fractional step procedure detailed later. Once the new pressure field is known, the velocity vector field at the new Picard iteration $\hat{\mathbf{v}}^{n+1,k+1}$ can be readily updated from the discrete momentum equation (32).

To close the problem it remains to specify how to construct the nonlinear convective-diffusion operator $\widehat{\boldsymbol{F}\mathbf{v}}_j^{n+1,k+\frac{1}{2}}$. At this point one can try to extend the procedure already used in [39] to 3D. However, in this case there are some issues that have to be taken into account. In particular, since we are using a modal basis on the staggered dual non-standard 5-point hexahedral mesh, we cannot use the simple nodal approximation for the nonlinear convective term $\hat{\mathbf{F}}_c = \mathbf{F}_c(\hat{\mathbf{v}})$ that consists in a trivial point-wise evaluation of the nonlinear operator $\mathbf{F}_c$. Inspired by the good properties obtained by the use of staggered grids, here we propose a new procedure for the computation of the nonlinear convective and viscous terms. For that purpose, the velocity field is first interpolated from the dual grid to the main grid. The nonlinear convective terms can then be easily discretized with a standard (space–time) DG scheme on the main grid. Then, the staggered mesh is used *again* in order to define the gradient of the velocity on the dual elements, which allows us to produce a very simple and sparse system for the discretization of the viscous terms.

An implicit discretization of the viscous terms on the dual grid leads to a linear system for each velocity component that is a seven-point non-symmetric block system that is well conditioned for convection dominated problems, for which it can be written as a ν perturbation of the identity matrix, see e.g. [39]. Here, we will develop a discretization of the viscous terms that leads only to a five-point block system and, more importantly, is symmetric and positive definite for $\nu > 0$ and $p_\gamma = 0$, but is still better conditioned also in the general case $p_\gamma > 0$.

Given a discrete velocity field $\mathbf{v}_h$ on the dual grid in the time interval $[t^n, t^{n+1}]$, we can project the velocity field from the dual mesh to the main grid (denoted by $\bar{\mathbf{v}}$) via standard L_2 projection,

$$\bar{\mathbf{v}}_i^{n+1} = \boldsymbol{M}_i^{-1}\sum_{j\in S_i}\boldsymbol{M}_{i,j}\hat{\mathbf{v}}_j^{n+1}, \qquad \forall i \in [1, N_e], \tag{35}$$

where $\bar{\mathbf{v}}_i^{n+1}$ denote the degrees of freedom of the velocity on the main grid and

$$\boldsymbol{M}_i = \int_{\boldsymbol{T}_i^{st}} \tilde{\phi}_k^{(i)}\tilde{\phi}_l^{(i)} d\mathbf{x}dt, \qquad \boldsymbol{M}_{i,j} = \int_{\boldsymbol{T}_{i,j}^{st}} \tilde{\phi}_k^{(i)}\tilde{\psi}_l^{(j)} d\mathbf{x}dt. \tag{36}$$

The projection back onto the dual grid is given by

$$\hat{\mathbf{v}}_j^{n+1} = \overline{\boldsymbol{M}}_j^{-1}\left(\boldsymbol{M}_{\ell(j),j}^{\top}\bar{\mathbf{v}}_{\ell(j)}^{n+1} + \boldsymbol{M}_{r(j),j}^{\top}\bar{\mathbf{v}}_{r(j)}^{n+1}\right), \tag{37}$$

with

$$\overline{\boldsymbol{M}}_j = \int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)}\tilde{\psi}_l^{(j)} d\mathbf{x}dt. \tag{38}$$

We can rewrite the nonlinear convective and viscous part of the momentum equation by introducing the viscous stress tensor $\boldsymbol{\sigma} = -\nu\nabla\mathbf{v}$ as auxiliary variable. The convective and viscous subsystem of the momentum equation then reads

$$\begin{aligned} &\frac{\partial \mathbf{v}}{\partial t} + \nabla\cdot\mathbf{F}_c + \nabla\cdot\boldsymbol{\sigma} = 0, \\ &\boldsymbol{\sigma} = -\nu\nabla\mathbf{v}. \end{aligned} \tag{39}$$

With the averaged velocity $\bar{\mathbf{v}}_i^{n+1} = \tilde{\phi}_l^{(i)} \bar{\mathbf{v}}_{l,i}^{n+1}$ defined on the main grid and the viscous stress tensor $\boldsymbol{\sigma}_j^{n+1} = \tilde{\psi}_l^{(j)} \boldsymbol{\sigma}_{l,j}^{n+1}$ defined on the dual grid, we obtain the following weak formulation of (39):

$$\int_{\boldsymbol{T}_i} \tilde{\phi}_k^{(i)}(\mathbf{x}, t^{n+1}) \bar{\mathbf{v}}_i^{n+1} \, d\mathbf{x} - \int_{\boldsymbol{T}_i} \tilde{\phi}_k^{(i)}(\mathbf{x}, t^{n}) \bar{\mathbf{v}}_i^{n} \, d\mathbf{x} - \int_{\boldsymbol{T}_i^{st}} \frac{\partial \tilde{\phi}_k^{(i)}}{\partial t} \bar{\mathbf{v}}_i^{n+1} \, d\mathbf{x}dt +$$

$$\int_{\partial \boldsymbol{T}_i^{st}} \tilde{\phi}_k^{(i)} \mathbf{F}_c^{\mathrm{RS}}\left(\bar{\mathbf{v}}^-, \bar{\mathbf{v}}^+\right) \cdot \vec{n}_i \, dSdt - \int_{\boldsymbol{T}_i^{st}} \nabla \tilde{\phi}_k^{(i)} \cdot \mathbf{F}_c(\bar{\mathbf{v}}_i^{n+1}) \, d\mathbf{x}dt + \sum_{j \in S_i} \left(\int_{\Gamma_j^{st}} \tilde{\phi}_k^{(i)} \boldsymbol{\sigma}_j^{n+1} \cdot \vec{n}_{ij} \, dSdt - \int_{\boldsymbol{T}_{i,j}^{st}} \nabla \tilde{\phi}_k^{(i)} \cdot \boldsymbol{\sigma}_j^{n+1} \, d\mathbf{x}dt \right) = 0, \tag{40}$$

$$\int_{\boldsymbol{H}_j^{st}} \tilde{\psi}_k^{(j)}(\mathbf{x}, t^{n+1}) \boldsymbol{\sigma}_j^{n+1} \, d\mathbf{x}$$

$$= -\nu \left(\int_{\boldsymbol{T}_{\ell(j),j}^{st}} \tilde{\psi}_k^{(j)} \nabla \bar{\mathbf{v}}_{\ell(j)}^{n+1} \, d\mathbf{x}dt + \int_{\boldsymbol{T}_{r(j),j}^{st}} \tilde{\psi}_k^{(j)} \nabla \bar{\mathbf{v}}_{r(j)}^{n+1} \, d\mathbf{x}dt + \int_{\Gamma_j^{st}} \tilde{\psi}_k^{(j)} \left(\bar{\mathbf{v}}_{r(j)}^{n+1} - \bar{\mathbf{v}}_{\ell(j)}^{n+1} \right) \otimes \vec{n}_j \, dSdt \right). \tag{41}$$

In a more compact matrix notation, (41) can be written as:

$$\left(\overline{\boldsymbol{M}}_i^{+} - \overline{\boldsymbol{M}}_i^{o} \right) \bar{\mathbf{v}}_i^{n+1} - \overline{\boldsymbol{M}}_i^{-} \bar{\mathbf{v}}_i^{n} + \sum_{j \in S_i} \boldsymbol{\mathcal{D}}_{i,j} \boldsymbol{\sigma}_j^{n+1} + \overline{\Upsilon}_i^{c} = 0,$$

$$\overline{\boldsymbol{M}}_j \boldsymbol{\sigma}_j^{n+1} = -\nu \left(\boldsymbol{\mathcal{Q}}_{\ell(j),j} \bar{\mathbf{v}}_{\ell(j)}^{n+1} + \boldsymbol{\mathcal{Q}}_{r(j),j} \bar{\mathbf{v}}_{r(j)}^{n+1} \right), \tag{42}$$

where

$$\overline{\boldsymbol{M}}_i^{+} = \int_{\boldsymbol{T}_i} \tilde{\phi}_k^{(i)}(\mathbf{x}, t(1)) \tilde{\phi}_l^{(i)}(\mathbf{x}, t(1)) d\mathbf{x}, \tag{43}$$

$$\overline{\boldsymbol{M}}_i^{-} = \int_{\boldsymbol{T}_i} \tilde{\phi}_k^{(i)}(\mathbf{x}, t(0)) \tilde{\phi}_l^{(i)}(\mathbf{x}, t(1)) d\mathbf{x}, \tag{44}$$

$$\overline{\boldsymbol{M}}_i^{o} = \int_{\boldsymbol{T}_i^{st}} \frac{\partial \tilde{\phi}_k^{(i)}}{\partial t} \tilde{\phi}_l^{(i)} d\mathbf{x}dt. \tag{45}$$

In (42) we have defined the operator $\overline{\Upsilon}_i^{c}(\bar{\mathbf{v}})$, which is a standard DG discretization of the nonlinear convective terms on the tetrahedral elements of the *main grid*,

$$\overline{\Upsilon}_i^{c}(\bar{\mathbf{v}}) = \int_{\partial \boldsymbol{T}_i^{st}} \tilde{\phi}_k^{(i)} \mathbf{F}_c^{\mathrm{RS}}\left(\bar{\mathbf{v}}^-, \bar{\mathbf{v}}^+\right) \cdot \vec{n}_i \, dSdt - \int_{\boldsymbol{T}_i^{st}} \nabla \tilde{\phi}_k^{(i)} \cdot \mathbf{F}_c(\bar{\mathbf{v}}) \, d\mathbf{x}dt, \tag{46}$$

with the boundary extrapolated values $\mathbf{v}^-$ and $\mathbf{v}^+$ from within the cell and from the neighbors, respectively. Here, the approximate Riemann solver $\mathbf{F}_c^{\mathrm{RS}}$ used at the element boundaries is given by the simple Rusanov flux [84]

$$\mathbf{F}_c^{\mathrm{RS}}\left(\bar{\mathbf{v}}^-, \bar{\mathbf{v}}^+\right) \cdot \vec{n}_i = \frac{1}{2} \left(\mathbf{F}_c(\bar{\mathbf{v}}^+) + \mathbf{F}_c(\bar{\mathbf{v}}^-) \right) \cdot \vec{n}_i - \frac{1}{2} s_{\max} \left(\bar{\mathbf{v}}^+ - \bar{\mathbf{v}}^- \right), \tag{47}$$

where $s_{\max} = 2 \max\left(|\bar{\mathbf{v}}^+|, |\bar{\mathbf{v}}^-| \right)$ is the maximum eigenvalue of the convective operator $\mathbf{F}_c$. The final system for the variable $\bar{\mathbf{v}}$ can be found by formal substitution of $\boldsymbol{\sigma}$ given in the second equation of (42) into the first one:

$$\left(\overline{\boldsymbol{M}}_i - \nu \sum_{j \in S_i} \boldsymbol{\mathcal{D}}_{i,j} \overline{\boldsymbol{M}}_j^{-1} \boldsymbol{\mathcal{Q}}_{i,j} \right) \bar{\mathbf{v}}_i^{n+1} - \nu \sum_{j \in S_i} \boldsymbol{\mathcal{D}}_{i,j} \overline{\boldsymbol{M}}_j^{-1} \boldsymbol{\mathcal{Q}}_{\wp(i,j),j} \bar{\mathbf{v}}_{\wp(i,j)}^{n+1} = \overline{\boldsymbol{M}}_i^{-} \bar{\mathbf{v}}_i^{n} - \overline{\Upsilon}_i^{c}(\bar{\mathbf{v}}^{n+1}), \tag{48}$$

where we use the abbreviation $\overline{\boldsymbol{M}}_i = \overline{\boldsymbol{M}}_i^{+} - \overline{\boldsymbol{M}}_i^{o}$. What we obtain is a discretization of the nonlinear convective and viscous terms on the main grid, where the stress tensor $\boldsymbol{\sigma}$ has been computed on the face-based dual mesh. In order to avoid the solution of a nonlinear system due to the nonlinear operator $\overline{\Upsilon}_i^{c}(\bar{\mathbf{v}}^{n+1})$, we introduce a fractional step scheme combined with an outer Picard iteration. Using the notation introduced in [39], we get

$$\left(\overline{\mathbf{M}}_i - \nu \sum_{j\in S_i} \mathcal{D}_{i,j} \overline{\mathbf{M}}_j^{-1} \mathcal{Q}_{i,j}\right) \overline{\mathbf{v}}_i^{n+1,k+\frac{1}{2}} - \nu \sum_{j\in S_i} \mathcal{D}_{i,j} \overline{\mathbf{M}}_j^{-1} \mathcal{Q}_{\wp(i,j),j} \overline{\mathbf{v}}_{\wp(i,j)}^{n+1,k+\frac{1}{2}} = \overline{\mathbf{M}}_i^{-} \overline{\mathbf{v}}_i^{n} - \overline{\Upsilon}_i^{c}\left(\overline{\mathbf{v}}^{n+1,k}\right). \tag{49}$$

It is important to emphasize that boundary conditions are enforced in the DG context only in a weak integral sense, which makes their implementation particularly simple. This is different with respect to other schemes, for which the issues with boundary conditions have been discussed in [89]. For example, solid wall boundaries require $\mathbf{v} = 0$ at the wall. In the nonlinear convective terms of our scheme this is simply achieved by setting at the boundary $\bar{\mathbf{v}}^{+} = -\bar{\mathbf{v}}^{-}$ in (47), where $\bar{\mathbf{v}}^{-}$ is the known boundary-extrapolated velocity from inside the computational domain. For the implementation of the wall boundary condition in the viscous terms, one just needs to set $\bar{\mathbf{v}}_{r(j)}^{n+1} = 0$ in (41) (assuming that $\ell(j)$ is the existing element inside the computational domain), and in the same equation the control volume for the computation of the stress tensor is simply changed from $\boldsymbol{H}_j^{st}$ to $\boldsymbol{T}_{\ell(j),j}^{st}$, since the contribution of $\boldsymbol{T}_{r(j),j}^{st}$ does not exist at the boundary. In this context we also would like to emphasize that the purpose of the Picard iteration in our algorithm is *not* to deal with boundary conditions, but just to get high order in time for $p_\gamma > 0$.

2.6. Final space–time pressure correction formulation

As already discussed in [39], the computation of the nonlinear convective and viscous terms presented in Eq. (49) does not depend explicitly on the pressure of the previous Picard iteration, and hence it does not see the effect of the pressure in the time interval T^{n+1}, which is, however, needed to get a high order accurate scheme also in time. In order to overcome the problem, we introduce directly into Eq. (49) the contribution of the pressure in the time interval T^{n+1}, but at the previous Picard iteration. Then, we update the velocity with the pressure correction $\hat{\mathbf{p}}_i^{n+1,k+1} - \hat{\mathbf{p}}_i^{n+1,k}$. The final equations (49), (32) and (33) to be solved for each Picard iteration k of our staggered semi-implicit space–time DG method therefore read:

$$\bar{\mathbf{v}}_i^{n+1,k} = \boldsymbol{M}_i^{-1} \sum_{j\in S_i} \boldsymbol{M}_{i,j} \hat{\mathbf{v}}_j^{n+1,k}, \tag{50}$$

$$\boldsymbol{\Lambda}_i(\hat{\boldsymbol{p}}^{n+1,k}) = \boldsymbol{M}_i^{-1} \sum_{j\in S_i} \boldsymbol{M}_{i,j} \left(\boldsymbol{M}_j^{-1} \left(\mathcal{Q}_{r(j),j} \hat{\mathbf{p}}_{r(j)}^{n+1,k} + \mathcal{Q}_{\ell(j),j} \hat{\mathbf{p}}_{\ell(j)}^{n+1,k}\right)\right), \tag{51}$$

$$\begin{aligned}&\left(\overline{\mathbf{M}}_i - \nu \sum_{j\in S_i} \mathcal{D}_{i,j} \overline{\mathbf{M}}_j^{-1} \mathcal{Q}_{i,j}\right) \overline{\mathbf{v}}_i^{n+1,k+\frac{1}{2}} - \nu \sum_{j\in S_i} \mathcal{D}_{i,j} \overline{\mathbf{M}}_j^{-1} \mathcal{Q}_{\wp(i,j),j} \overline{\mathbf{v}}_{\wp(i,j)}^{n+1,k+\frac{1}{2}} \\ &= \overline{\mathbf{M}}_i^{-} \overline{\mathbf{v}}_i^{n} - \overline{\Upsilon}_i^{c}\left(\overline{\mathbf{v}}^{n+1,k}\right) - \overline{\mathbf{M}}_i \boldsymbol{\Lambda}_i(\hat{\boldsymbol{p}}^{n+1,k}),\end{aligned} \tag{52}$$

$$\widehat{\boldsymbol{F}}\mathbf{v}_j^{n+1,k+\frac{1}{2}} = \overline{\mathbf{M}}_j^{-1} \left(\boldsymbol{M}_{\ell(j),j}^{\top} \overline{\mathbf{v}}_{\ell(j)}^{n+1,k+\frac{1}{2}} + \boldsymbol{M}_{r(j),j}^{\top} \overline{\mathbf{v}}_{r(j)}^{n+1,k+\frac{1}{2}}\right), \tag{53}$$

$$\sum_{j\in S_i} \mathcal{D}_{i,j} \boldsymbol{M}_j^{-1} \mathcal{Q}_{i,j} \left(\hat{\mathbf{p}}_i^{n+1,k+1} - \hat{\mathbf{p}}_i^{n+1,k}\right) + \sum_{j\in S_i} \mathcal{D}_{i,j} \boldsymbol{M}_j^{-1} \mathcal{Q}_{\wp(i,j),j} \left(\hat{\mathbf{p}}_{\wp(i,j)}^{n+1,k+1} - \hat{\mathbf{p}}_{\wp(i,j)}^{n+1,k}\right) = \sum_{j\in S_i} \mathcal{D}_{i,j} \widehat{\boldsymbol{F}}\mathbf{v}_j^{n+1,k+\frac{1}{2}}, \tag{54}$$

$$\hat{\mathbf{v}}_j^{n+1,k+1} = \widehat{\boldsymbol{F}}\mathbf{v}_j^{n+1,k+\frac{1}{2}} - \boldsymbol{M}_j^{-1} \left(\mathcal{Q}_{r(j),j} \left(\hat{\mathbf{p}}_{r(j)}^{n+1,k+1} - \hat{\mathbf{p}}_{r(j)}^{n+1,k}\right) + \mathcal{Q}_{\ell(j),j} \left(\hat{\mathbf{p}}_{\ell(j)}^{n+1,k+1} - \hat{\mathbf{p}}_{\ell(j)}^{n+1,k}\right)\right), \tag{55}$$

where $\boldsymbol{\Lambda}_i(\hat{\boldsymbol{p}}^{n+1,k})$ represents the same additional contribution subtracted in (54) that lives on the dual mesh, passed through the averaging operator from the dual to the main grid. As initial guess for the pressure we simply take $\hat{\boldsymbol{p}}^{n+1,0} = 0$, while for the velocity field we simply take the velocity field at the previous time step. As an alternative, one could also take an extrapolation of pressure and velocity from the previous time interval. A summary of the algorithm reads:

0. Choose an initial guess for the pressure and the velocity.
1. average the velocity field from the dual grid to the main grid using (50) and compute the contribution of the pressure gradient of the previous Picard iteration on the main grid using (51);
2. with the averaged velocity on the main grid, compute the nonlinear convective terms via (46);
3. solve the linear systems for the viscous terms (52) on the main grid;
4. compute the term $\widehat{\boldsymbol{F}}\mathbf{v}_j^{n+1,k+\frac{1}{2}}$ on the dual grid via (53);
5. solve the linear system for the pressure correction (54) on the main grid;
6. update the velocity field according to (55) using the previously obtained pressure correction.

Steps 1–6 are repeated for a total number of Picard iterations of $N_{pic} = p_\gamma + 1$, which is inspired by the behavior of the Picard process applied to ODE, where it directly constructs the Taylor series expansion of the solution and thus allows to gain one order of accuracy per iteration. Furthermore, based on our previous experience gathered with local space–time

Galerkin methods used in high order one-step ADER schemes [90,19,91], and based on our numerical results presented in Section 3, we *conjecture* that this property also holds for the above algorithm. Other numerical schemes that follow a similar approach are the so-called deferred correction algorithms, see e.g. [92,93]. We believe that the key ingredient of our algorithm, which allows the simple Picard iteration to work, is the use of a space–time DG formalism that represents the discrete solution also in terms of high order polynomials in time.

2.7. Remarks on the special case of piecewise constant polynomials in time ($p_\gamma = 0$)

Apart from the new treatment of the viscous terms introduced in this paper, the method presented in the previous sections can be seen, for $p_\gamma = 0$, as the extension of [61] to three space dimensions. This particular case is, in general, only first order accurate in time but high order accurate in space. We also stress that for $p_\gamma = 0$, *no* Picard iterations are used, since the Picard process is only necessary to obtain higher order in time, but keeping at the same time the structure of the entire scheme simple. In this case, we can recover several good properties for the main system for the pressure and for the linear systems that need to be solved for the implicit discretization of the viscous terms.

2.7.1. Pressure system

For $p_\gamma = 0$ we have $\boldsymbol{M}_j^\circ = 0$ then $\boldsymbol{M}_j = \boldsymbol{M}_j^+ = \boldsymbol{M}_j^-$ is symmetric for all $j \in 1 \dots N_d$. Consequently, the system (31)–(32) formally becomes the same method as in [61]. The following results can therefore be readily obtained as corollaries of the theorems given in [61] regarding the system matrix $\mathcal{A}$ of the main system for the pressure (33):

Corollary 1 *(Symmetry). Let $p_\gamma = 0$, the system matrix $\mathcal{A}$ of the main system for the pressure is symmetric.*

Corollary 2 *(Positive semi-definiteness). Let $p_\gamma = 0$, the system matrix $\mathcal{A}$ of the main system for the pressure is in general positive semi-definite.*

This means that in this particular case we can use faster iterative linear solvers, like the conjugate gradient (CG) method [52] to solve the main system for the pressure (33). This advantage makes the case $p_\gamma = 0$ particularly suitable for steady or almost steady problems. In order to recover some precision in time we can extend the algorithm by introducing a semi-implicit discretization, as suggested in [61]. In this case, system (31)–(32) has to be discretized as

$$\sum_{j \in S_i} \mathcal{D}_{i,j} \hat{\mathbf{v}}_j^{n+1} = 0, \tag{56}$$

$$\boldsymbol{M}_j \hat{\mathbf{v}}_j^{n+1} - \boldsymbol{M}_j \widehat{\boldsymbol{F}\mathbf{v}}_j^{n} + \Delta t \mathcal{Q}_{r(j),j} \hat{\mathbf{p}}_{r(j)}^{n+\theta} + \Delta t \mathcal{Q}_{\ell(j),j} \hat{\mathbf{p}}_{\ell(j)}^{n+\theta} = 0, \tag{57}$$

where $\hat{\mathbf{p}}^{n+\theta} = \theta \hat{\mathbf{p}}^{n+1} + (1-\theta)\hat{\mathbf{p}}^n$ and θ is an implicitness factor to be taken in the range $\theta \in [\frac{1}{2}, 1]$, see e.g. [94]. For $\theta = \frac{1}{2}$, the Crank–Nicolson method is recovered. In this way we gain some extra precision in time without affecting the computational effort and using the same advantages given by Corollary 1 and 2 that can be easily extended for this case.

2.7.2. Viscous system

In the special case of piecewise constant polynomials in time ($p_\gamma = 0$), we get $\overline{\boldsymbol{M}}_i = \boldsymbol{M}_i$ and $\overline{\boldsymbol{M}}_j = \boldsymbol{M}_j$, so that the following results about the viscous system (49) can be derived:

Corollary 3 *(Symmetry). If $p_\gamma = 0$ then the system (49) is symmetric.*

Proof. We can write the system matrix of system (49) as $(M + \nu\mathcal{A})$, where M is a block diagonal matrix with $\{\boldsymbol{M}_i\}_{i=1\dots N_e}$ on the diagonal and $\mathcal{A}$ is the matrix of the pressure system (33). Thanks to the results obtained in Corollary 1, $\mathcal{A}$ is symmetric and also M is symmetric, since $\boldsymbol{M}_i = \boldsymbol{M}_i^\top$, see (43). □

Corollary 4 *(Positive definiteness). If $p_\gamma = 0$ then the system (49) is positive definite.*

Proof. As used in Corollary 3, we can write the system such as $M + \nu\mathcal{A}$ and we know, thanks to Corollary 2, that $\mathcal{A}$ is in general positive semi-definite. A simple computation leads to

$$x(M + \nu\mathcal{A})x^\top = xMx^\top + \nu x\mathcal{A}x^\top > 0 \tag{58}$$

since $\nu x\mathcal{A}x^\top \geq 0$ and $xMx^\top > 0$ we have that the complete system is also positive definite. □

In the general case of $p_\gamma > 0$ it is not true that we recover the pressure system, since $\overline{\boldsymbol{M}}_\cdot \neq \boldsymbol{M}_\cdot$. In this case, we can observe how the non-symmetric contribution affects only $\overline{\boldsymbol{M}}_i$. This allows us to write the previous system as $T + \nu H$ where T is a block diagonal non-symmetric matrix and H is symmetric and positive semi-definite.

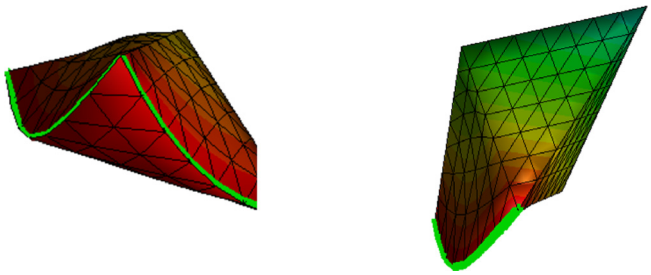

Fig. 3. Isoparametric dual element example. On the left, a dual element with a 2D face on the curved boundary; on the right, an internal dual element, but with a 1D edge on the curved boundary.

2.8. Extension to curved elements

The method described in the previous sections can readily be generalized by introducing also curved elements inside the computational domain following an iso-parametric approach. This generalization will affect only the pre-processing step. The extension is quite similar to the one introduced in [39,60] for the two dimensional case, but there are some differences due to the three dimensionality of the problem.

First of all, in the two dimensional case one could eventually consider as curved only the primary elements that touch a curved boundary, as well as the associated dual elements such that $j \in \mathcal{B}(\Omega)$. In the $3D$ case we have to curve also those internal elements which touch the boundary with an edge, see for example Fig. 3. Each tetrahedral main element is then characterized by N_ϕ nodes $\{(X,Y,Z)_k\}_{k=1,N_\phi}$, while the dual hexahedral elements are split into a left and a right tetrahedron, i.e. $\boldsymbol{H}_j^{iso} = \boldsymbol{T}_{\ell(j),j}^{iso} \cup \boldsymbol{T}_{r(j),j}^{iso}$ and the points that lie on Γ_j^{iso} are physically joined. In this way we have a full characterization of the left and the right sub-tetrahedron of the dual hexahedral element, needed to compute properly the integral contributions in the algorithm.

In order to compute the position of the grid points in the presence of curved boundaries, we start from an initial tetrahedrization with piecewise linear faces, as given by a standard mesh generator. Then, we produce a fine sub-tetrahedrization that involves all the degrees of freedom inside the domain Ω and we solve a simple Laplace equation for the displacement using a classical $P1$ continuous finite element method, imposing the projection onto the curved physical boundaries as boundary conditions for the Laplace equation. This procedure produces a regular distribution of nodes inside the computational domain in the presence of curved boundaries.

As shown in [39], the possibility to curve the grid is crucial when we try to represent complex domains with a very coarse grid. In any case, we emphasize that this generalization does not affect the computational cost during run-time, since it affects only the construction of the main matrices that can be done in a preprocessing step.

3. Numerical test problems

3.1. Three-dimensional lid driven cavity

In this section we present some results regarding the three-dimensional lid-driven cavity problem. In the literature there are a lot of well known results and reference solutions for the two-dimensional as well as for the fully three-dimensional case, see [95–97,46,47]. We take a classical cubic cavity $\Omega = [-0.5, 0.5]^3$ and we discretize it with a very coarse tetrahedral mesh with characteristic mesh size $h = 0.2$. We set as initial conditions $p = 1$; $u = v = w = 0$. As boundary condition we impose velocity $(u, v, w) = (1, 0, 0)$ at $y = 0.5$ while no-slip boundary conditions are used on the other boundaries. Since we are interested in steady state solutions, we take for the current test $p = 4$, $p_\gamma = 0$, $\theta = 1$, and several different values for the kinematic viscosity in order to obtain different Reynolds numbers.

In Fig. 4 the results are shown at a final time of $t_{end} = 30$ for $Re = 400$. In Fig. 5 the same plots are given for $t_{end} = 40$ and $Re = 1000$. In the top left panel of each plot we report our numerical results and compare them against the reference solution obtained in [97] for the fully three-dimensional case and the data given by Ghia et al. [95] for the two dimensional cavity at the same Reynolds number. We note a very good agreement with the 3D reference solution, despite the use of an extremely coarse mesh. The data show that the presence of the third space dimension significantly modifies the velocity profiles compared to the 2D case. Furthermore, several Taylor–Görtler like vortices appear in the secondary planes in a very similar way as observed in other numerical and experimental investigations of this problem, see e.g. [46,47].

3.2. Convergence test

In this test we investigate the Arnold–Beltrami–Childress flow that was originally introduced by Arnold in [98] and Childress in [99] as an interesting class of Beltrami flows and successively studied in a series of papers, see e.g. [100–103]. In particular we consider:

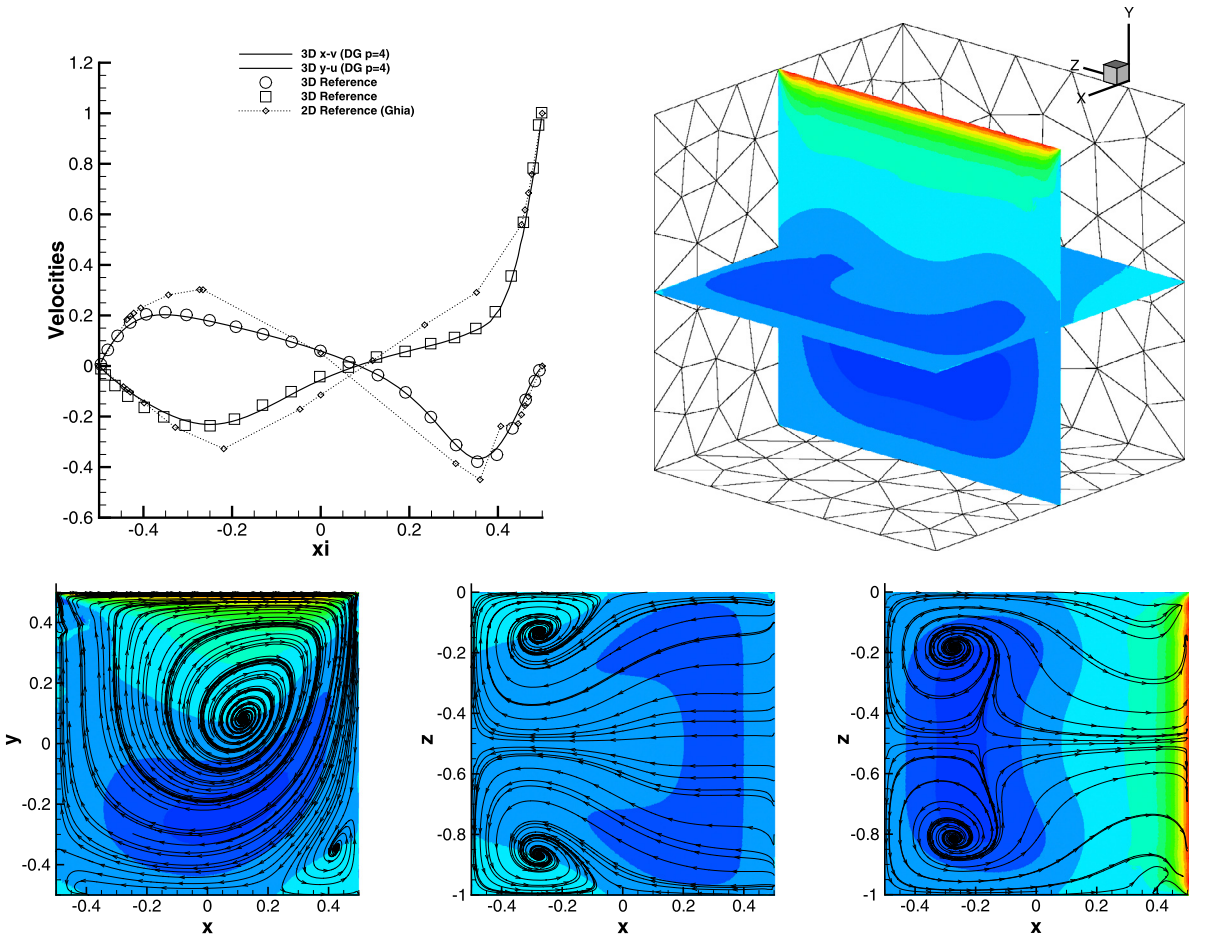

Fig. 4. 3D lid-driven cavity. From top left to bottom right: Comparison between our numerical results, the one obtained by Albensoeder et al. in [97], and the two dimensional data from Ghia et al. [95] at $Re = 400$; three-dimensional plot of the two secondary slices and grid spacing; streamlines and magnitude of u on slices x–y, x–z and y–z.

$$
\begin{aligned}
u(x, y, z, t) &= [\sin(z) + \cos(y)]\, e^{-\nu t}, \\
v(x, y, z, t) &= [\sin(x) + \cos(z)]\, e^{-\nu t}, \\
w(x, y, z, t) &= [\sin(y) + \cos(x)]\, e^{-\nu t}, \\
p(x, y, z, t) &= -[\cos(x)\sin(y) + \sin(x)\cos(z) + \sin(z)\cos(y)]\, e^{-2\nu t} + c
\end{aligned} \tag{59}
$$

where $c \in \mathbb{R}$. One can check that this is an exact solution for the complete three dimensional incompressible Navier–Stokes equations in a periodic domain, so this smooth configuration is suitable for numerical convergence tests. In particular if $\nu = 0$ we can check the accuracy of the spatial part of the algorithm, i.e. $p_\gamma = 0$, since the solution is a steady one. We take as computational domain $\Omega = [-\pi, \pi]^3$ and we extend it using periodic boundary conditions everywhere. We use increasing values of the polynomial degree p and use a sequence of successively refined meshes, starting from a regular initial mesh. Simulations are performed up to $t_{end} = 0.1$. The time step Δt is chosen according to the CFL time restriction for explicit DG schemes based on the magnitude of the flow velocity. Since we have periodic boundary conditions everywhere, we have a set of solutions for the pressure given by (59) up to a constant. In order to verify that also the pressure field is correct, we choose c in (59) a posteriori according to the mean value of the resulting numerical pressure.

The resulting vorticity, pressure and streamlines are plotted in Fig. 6, while in Table 1 the resulting L_2 error norms are reported for the steady case $\nu = 0$. We observe how the optimal order of convergence is obtained for this steady problem for the pressure, while a suboptimal order of convergence can be observed for the velocity field.

In the second test case we turn on the viscosity in order to make the problem unsteady. For this kind of problem we use the space–time DG implementation of the algorithm and we set the number of Picard iterations to $N_{pic} = p_\gamma + 1$. Unfortunately, as soon as we use a high order polynomial in time, the resulting pressure system in the form as it is written in this paper looses the symmetry property and hence we have to use a slower linear solver, such as the GMRES method. However, very recent results obtained by Fambri and Dumbser [104] after the submission of this paper show that it *actually*

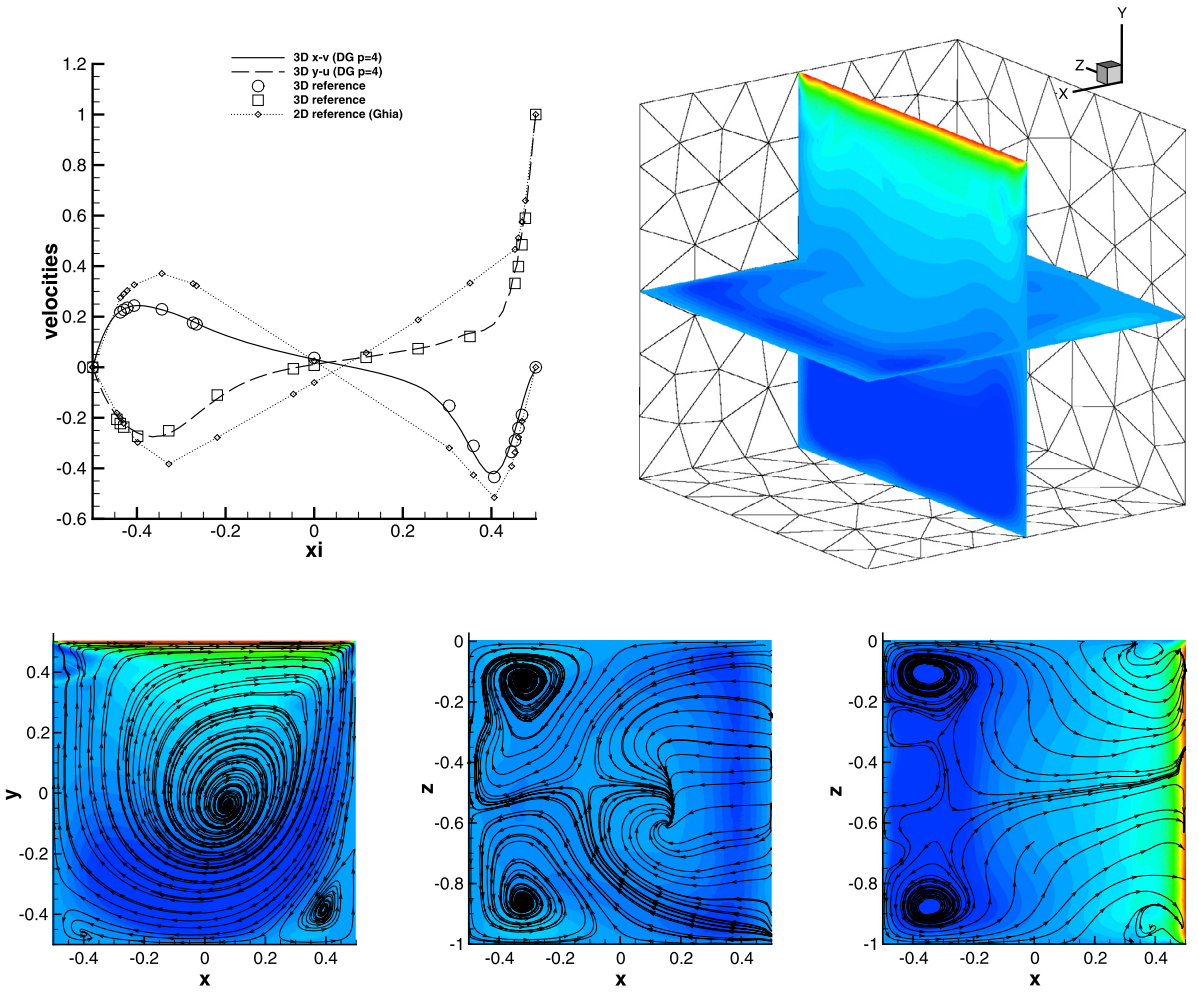

Fig. 5. 3D lid-driven cavity. From top left to bottom right: Comparison between our numerical results, the one obtained by Albensoeder et al. in [97], and the two dimensional data from Ghia et al. [95] at $Re = 1000$; three-dimensional plot of the two secondary slices and grid spacing; streamlines and magnitude of u on slices x–y, x–z and y–z.

is possible to *restore the symmetry* of the pressure system for staggered space–time DG schemes even for $p_\gamma > 0$, i.e. even for high order in time. Since the viscosity contribution is discretized implicitly, we can take very large values for the kinematic viscosity and maintain the same CFL time restriction for the simulation. The chosen viscosity for this test is $\nu = 1$ and we test the method for $p = p_\gamma = 1 \ldots 4$ on a sequence of successively refined grids. The resulting convergence rates, as well as the L_2 error norms, are shown in Table 2. In this case an order of $p + \frac{1}{2}$ is achieved for the pressure, while order $p + 1$ can be observed for the velocity.

3.3. Taylor–Green vortex

In this section we investigate another typical benchmark problem, namely the classical 3D Taylor–Green vortex. In this test case a very simple initial analytical solution degenerates quickly to a turbulent flow with very complex flow structures. We take the initial condition as given in [105]:

$$
\begin{aligned}
u(x, y, z, t) &= \sin(x)\cos(y)\cos(z), \\
v(x, y, z, t) &= -\cos(x)\sin(y)\cos(z), \\
w(x, y, z, t) &= 0, \\
p(x, y, z, t) &= p_0 + \frac{1}{16}\left(\cos(2x) + \cos(2y)\right)\left(\cos(2z) + 2\right),
\end{aligned}
\tag{60}
$$

Table 1
Numerical convergence results for the steady 3D ABC flow ($\nu = 0$).

p	p_γ	N_e	$\epsilon(p)$	$\epsilon(\mathbf{v})$	$\sigma(p)$	$\sigma(\mathbf{v})$
1	0	7986	7.4349E–01	3.7768E–01	–	–
1	0	10368	6.2638E–01	3.1662E–01	2.0	2.0
1	0	13182	5.3318E–01	2.7046E–01	2.0	2.0
1	0	16464	4.6155E–01	2.3309E–01	2.0	2.0
2	0	7986	8.6472E–02	5.0920E–02	3.0	2.4
2	0	10368	6.7178E–02	4.1417E–02	2.9	2.4
2	0	13182	5.2651E–02	3.4271E–02	3.0	2.4
2	0	16464	4.2520E–02	2.8499E–02	2.9	2.5
3	0	7986	6.6133E–03	3.5899E–03	3.9	3.4
3	0	10368	4.7069E–03	2.6619E–03	3.9	3.4
3	0	13182	3.4219E–03	2.0294E–03	4.0	3.4
3	0	16464	2.5604E–03	1.5727E–03	3.9	3.4
4	0	6000	8.4806E–04	6.7156E–04	4.9	4.1
4	0	7986	5.3156E–04	4.5361E–04	4.9	4.1
4	0	10368	3.4667E–04	3.1585E–04	4.9	4.2
4	0	13182	2.3307E–04	2.2733E–04	5.0	4.1
5	0	4374	1.5777E–04	1.6300E–04	5.9	5.1
5	0	6000	8.4744E–05	9.4463E–05	5.9	5.2
5	0	7986	4.8228E–05	5.7433E–05	5.9	5.2
5	0	10368	2.8868E–05	3.6318E–05	5.9	5.2

Table 2
Numerical convergence results for the unsteady ABC flow ($\nu = 1$).

p	p_γ	N_e	$\epsilon(p)$	$\epsilon(\mathbf{v})$	$\sigma(p)$	$\sigma(\mathbf{v})$
1	1	10368	1.1713E+00	2.4695E–01	1.6	2.0
1	1	13182	1.0388E+00	2.1017E–01	1.5	2.0
1	1	16464	9.2718E–01	1.8075E–01	1.5	2.0
1	1	20250	8.3860E–01	1.5730E–01	1.5	2.0
2	2	10368	1.7339E–01	1.4475E–02	2.8	3.1
2	2	13182	1.4060E–01	1.1291E–02	2.6	3.1
2	2	16464	1.1470E–01	8.9676E–03	2.8	3.1
2	2	20250	9.5780E–02	7.2516E–03	2.6	3.1
3	3	6000	1.6219E–02	1.5469E–03	3.8	4.1
3	3	7986	1.1454E–02	1.0494E–03	3.7	4.1
3	3	10368	8.2191E–03	7.3591E–04	3.8	4.1
3	3	13182	6.1399E–03	5.3142E–04	3.6	4.1
4	4	750	4.5578E–02	3.2574E–03	4.7	4.8
4	4	1296	1.9664E–02	1.2957E–03	4.6	5.1
4	4	2058	9.3757E–03	5.9049E–04	4.8	5.1
4	4	3072	5.0553E–03	2.9738E–04	4.6	5.1

in $\Omega = [\pi, \pi]^3$ and periodic boundary conditions everywhere. As numerical parameters we take $(p, p_\gamma) = (4, 0)$; $N_i = 494592$ tetrahedral elements; $\theta = 0.51$; Δt according to the CFL time restriction; $t_{end} = 10$; and several values of ν so that the Reynolds numbers under consideration are $Re = 400$, $Re = 800$ and $Re = 1600$, respectively.

A plot of the time evolution of the pressure field, the velocity magnitude and the vorticity pattern is shown in Figs. 7 and 8 for several times, as well as time series of the total kinematic energy dissipation rates compared with available DNS data given by Brachet et al. in [48] in Fig. 9. A good agreement between reference data and our numerical results can be observed. In Fig. 7 the vorticity pattern shows a really complex behavior that appears after a certain time (Fig. 8).

In this particular test it is very important to resolve the small scale structures that, close to $t = 9$, constitute the main contribution to the total kinetic energy dissipation. The mean number of iterations needed to solve the linear system for the pressure at $Re = 1600$ and a tolerance of $tol = 10^{-8}$ is $I_{mean} = 290.7$. In general we observe a number of iterations of the linear solver in the range $I \in [93, 2516]$ for this test case, without the use of any preconditioner.

3.4. Womersley flow

In this section the proposed algorithm is verified against the exact solution for an oscillating flow in a rigid tube of length L with circular cross section of diameter D. The unsteady flow is driven by a sinusoidal pressure gradient on the inlet and outlet boundaries

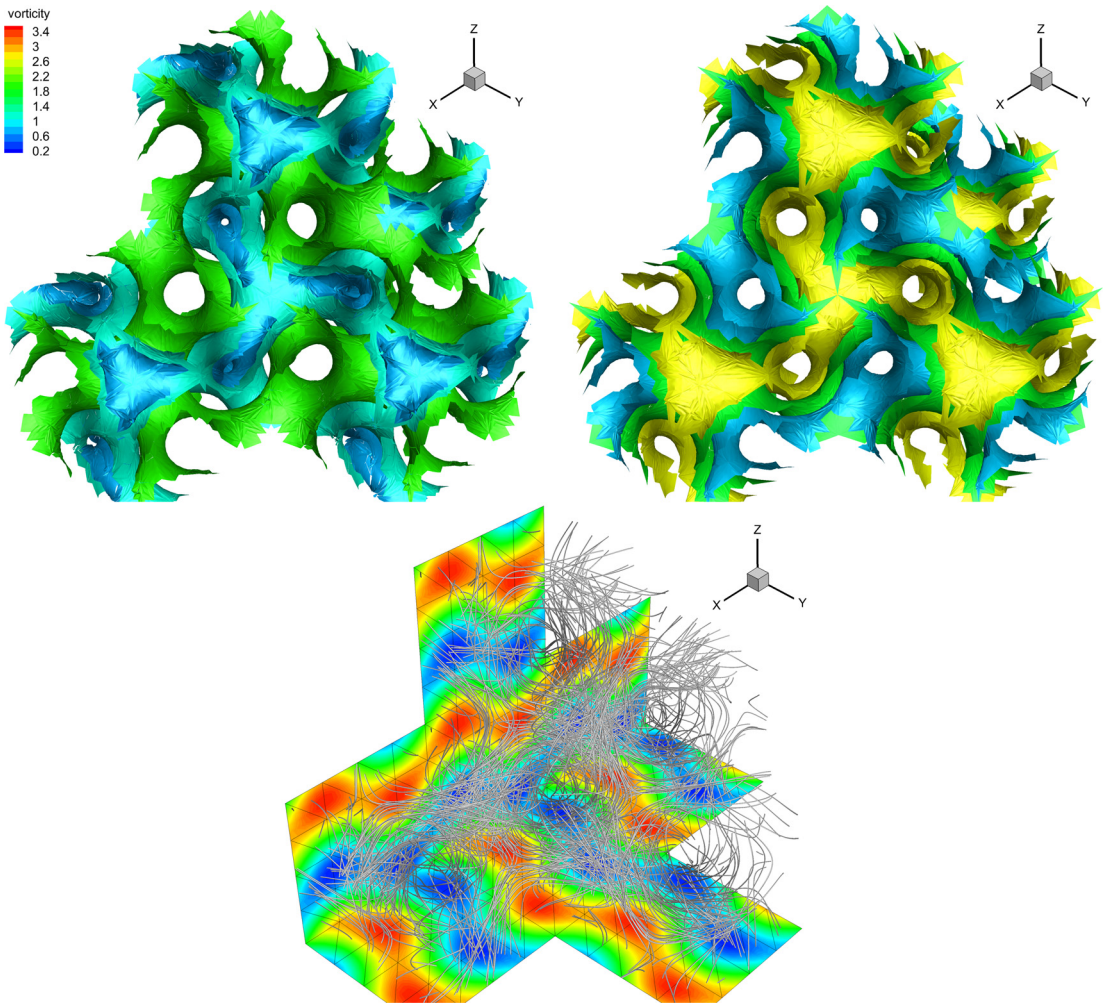

Fig. 6. From top left to bottom: Vorticity isosurfaces [0.8, 1.2, 2.0]; pressure isosurfaces $p = [-0.8, 0.0, 0.8]$ and streamlines in order to show the three-dimensionality of the ABC flow problem.

$$\frac{p_{out}(t) - p_{inlet}(t)}{L} = \frac{\tilde{p}}{\rho} e^{i\omega t}, \tag{61}$$

where $\tilde{p}$ is the amplitude of the pressure gradient; ρ is the fluid density; ω is the frequency of the oscillation; i indicates the imaginary unit; p_{inlet} and p_{out} are the inlet and outlet pressures, respectively. The analytical solution was derived by Womersley in [106]. According to [106,107] no convective contribution is considered. By imposing Eq. (61) at the tube ends, the resulting unsteady velocity field is uniform in the axial direction and is given by

$$u_e(\mathbf{x}, t) = \frac{\tilde{p}}{\rho} \frac{1}{i\omega} \left[1 - \frac{J_0\left(\alpha \zeta i^{\frac{3}{2}}\right)}{J_0\left(\alpha i^{\frac{3}{2}}\right)} \right] e^{i\omega t} \; ; \; v_e(\mathbf{x}, t) = w_e(\mathbf{x}, t) = 0, \tag{62}$$

where $\zeta = 2r/D$ with $r = \sqrt{y^2 + z^2}$ is the dimensionless radial coordinate; D is the diameter of the tube; $\alpha = \frac{D}{2}\sqrt{\frac{\omega}{\nu}}$ is a constant; and J_0 is the zero-th order Bessel function of the first kind. For the present test we take Ω as a cylinder (aligned with the x-axis) of length $L = 4$ and diameter $D = 2$; $\tilde{p} = 1000$; $\rho = 1000$; $\omega = 2\pi$; and $\nu = 0.04$. The computational domain Ω is covered with a total number of only $N_e = 1185$ tetrahedra and the time step size is chosen as $\Delta t = 0.3$, which is 30% of one oscillation period. For this test we take $(p, p_\gamma) = (4, 3)$ in order to produce a good solution also with the chosen time step Δt, which can be considered as very large for this problem.

Due to the curved geometry of the problem we use a fully isoparametric approach to fit the cylinder. A plot of the isoparametric grid that has been used here is reported in Fig. 12 on the left. We test our numerical solution in the cutting

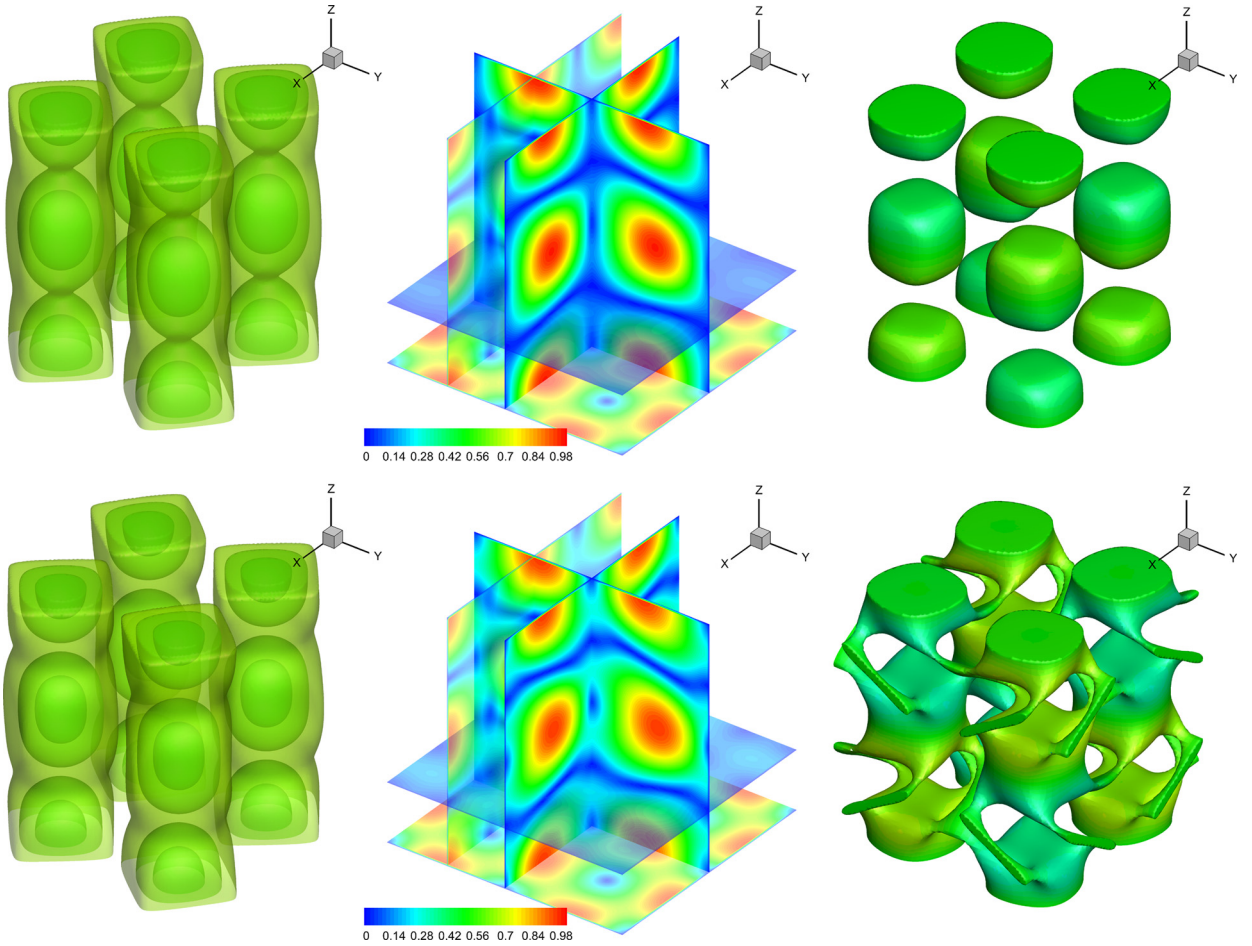

Fig. 7. 3D Taylor–Green vortex at $Re = 800$. From left to right: Pressure isosurfaces, velocity magnitude and vorticity isosurfaces at times $t = 0.5$ (top) and $t = 1.0$ (bottom).

slice $\Gamma = \{x = 2\}$ and successively on the line given by $(x, z) = (2, 0) \in \Gamma$. Fig. 10 shows the evolution of the velocity profile u on Γ solved in a single time cell $\Gamma^{st}(t, \vec{x}) = \Gamma(\vec{x}) \times [0.3, 0.6]$ evaluated at several intermediate times. A comparison between numerical and exact solution is reported in Fig. 11 as well as the plot of Γ, in order to show the axial symmetry of the solution, that is not trivial to obtain for the chosen discretization (very coarse unstructured mesh and very large time steps). Finally, a plot of the time series of the velocity u computed in $\mathbf{x} = (1, 0, 0)$ and $\mathbf{x} = (1, 0, 0.9)$ is reported in Fig. 12 and is compared with the exact solution. It is clear from Figs. 11 and 12 that this test with the chosen time step can reproduce good results only if we use high order polynomials also in time; indeed, the solution for a first order method in time would look piecewise constant within each time step.

3.5. Blasius boundary layer

We consider here a classical benchmark for viscous incompressible fluids. For the particular case of laminar stationary flow over a flat plate, a solution of Prandtl's boundary layer equations was found by Blasius in [108] and is given by the solution of a nonlinear third-order ODE, namely:

$$\begin{cases} f''' + ff'' = 0, \\ f(0) = 0, \quad f'(0) = 0, \quad \lim\limits_{\xi \to \infty} f'(\xi) = 1, \end{cases} \tag{63}$$

where $\xi = y\sqrt{\frac{u_\infty}{2\nu x}}$ is the Blasius coordinate; $f' = \frac{u}{u_\infty}$; and u_∞ is the far field velocity. The reference solution is computed here using a tenth-order DG ODE solver, see e.g. [19], together with a classical shooting method. In order to obtain the Blasius velocity profile in our simulations we consider a steady flow over a flat plate. As a result of the viscosity, a boundary layer appears along the no-slip wall. For the current test, we consider $\Omega = [-0.2, 0.8] \times [-0.2, 0.2]^2$. An initially uniform

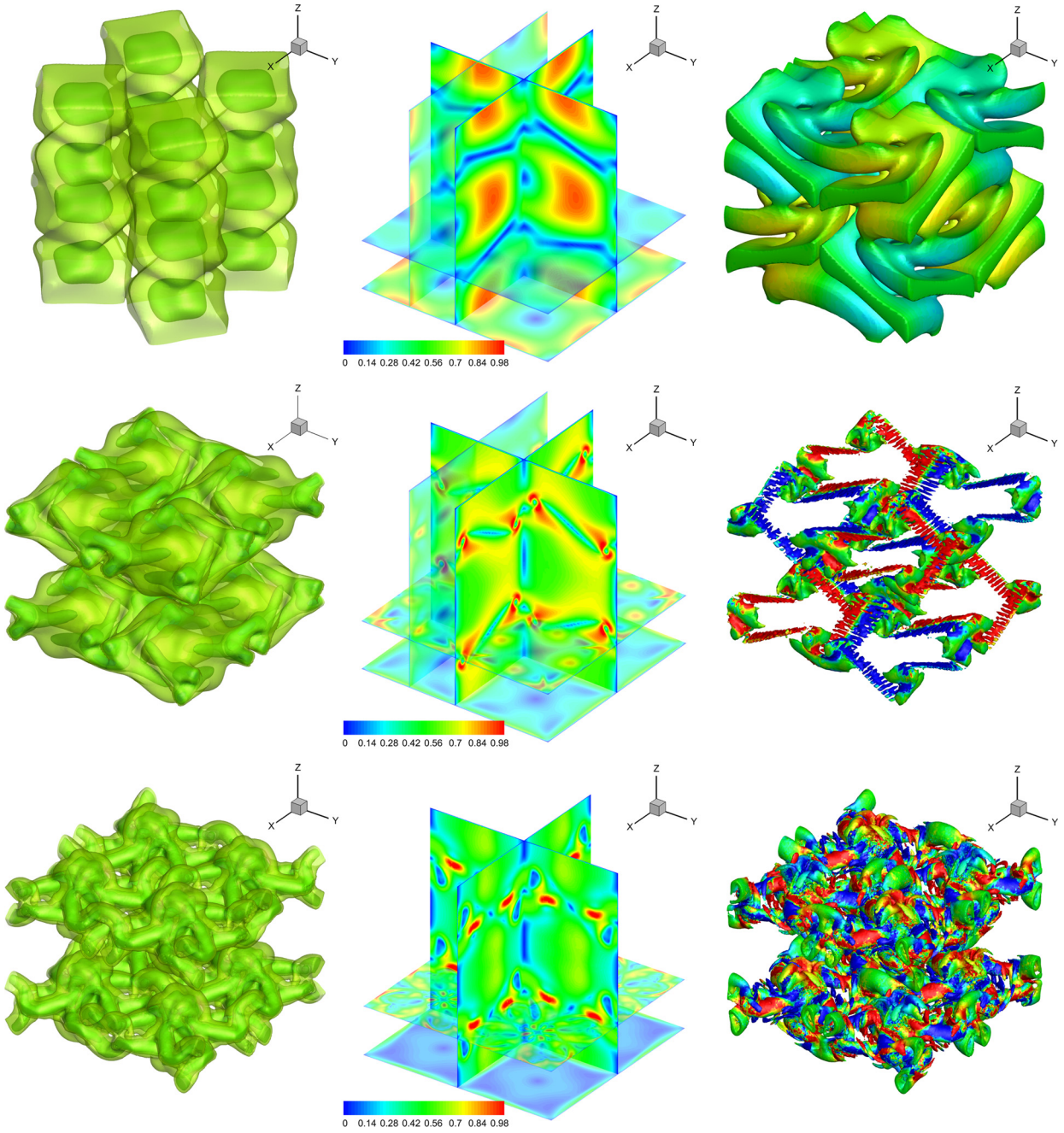

Fig. 8. 3D Taylor–Green vortex at $Re = 800$. From left to right: Pressure isosurfaces, velocity magnitude and vorticity isosurfaces at times $t = 2.1$ (top), $t = 4.8$ (center) and $t = 9.0$ (bottom).

flow $u(x, y, z, 0) = u_\infty = 1$, $v(x, y, z, 0) = w(x, y, z, 0) = 0$ and $p(x, y, 0) = 1$ is imposed as initial condition, while an inflow boundary is imposed on the left boundary; no slip boundary condition is considered in the flat plane $\Gamma = \{(x, y, z) | x \geq 0$ $y = y_{\min}\}$; slip boundary conditions are imposed at $z = z_{\min}$ and $z = z_{\max}$ and transmissive boundary conditions are imposed at the upper face $y = y_{\max}$. We consider here an extreme case of a very coarse mesh, where we cover our domain Ω with a set of only $N_e = 1522$ tetrahedra, whose characteristic length is $h = 0.07$. The chosen polynomial degree is $(p, p_\gamma) = (4, 0)$, the final simulation time is $t_{end} = 10$ and the viscosity is $\nu = 3 \cdot 10^{-4}$.

The resulting Blasius velocity profile is shown in Fig. 13 where also a sketch of the grid is reported. A comparison between the numerical results presented here and the Blasius solution is depicted in Fig. 14. A very good agreement between numerical and reference solution can be observed, which is quite remarkable, if we take into account the mesh size and considering that the major part of the boundary layer is essentially resolved in only one single control volume.

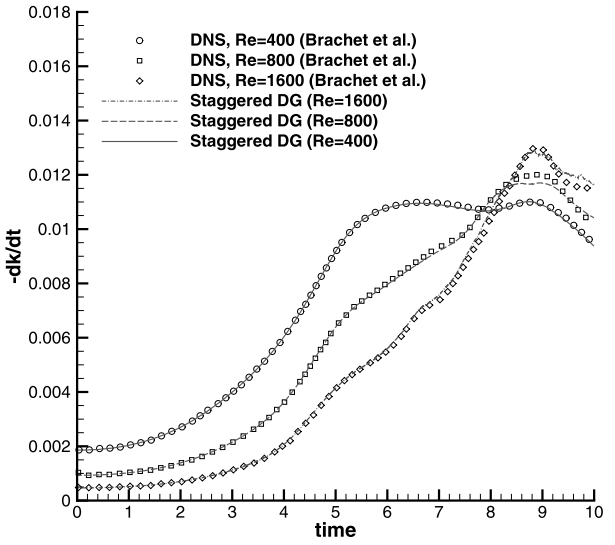

Fig. 9. Time evolution of the kinetic energy dissipation rate $-dk/dt$ for the 3D Taylor–Green vortex, compared with available DNS data of Brachet et al. [48] for $Re = 400, 800$ and $Re = 1600$.

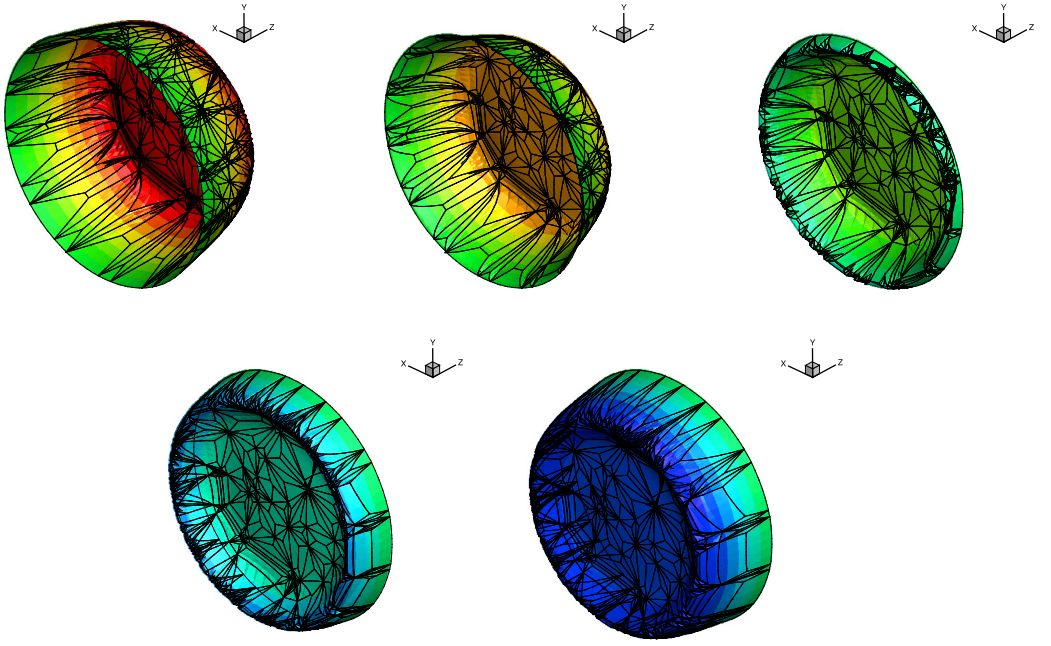

Fig. 10. 3D Womersley flow. Plot of u in the middle of the tube in *one single* time control volume $T = [0.3, 0.6]$. From top left to bottom right we plot the discrete solution at intermediate time levels $t = [0.3, 0.375, 0.45, 0.525, 0.6]$.

3.6. Backward-facing step

In this section, the three-dimensional numerical solution for the fluid flow over a backward-facing step is considered. For this test problem, both experimental and numerical results are available at several Reynolds numbers, see e.g. [45,109]. In particular, it is known that two dimensional simulations are in good agreement with experimental evidence only up to $Re = 400$. Beyond this critical value, two dimensional simulations present a large secondary recirculation zone that reduces the main recirculation zone. On the contrary, experimental results show that this secondary vortex appears only at higher Reynolds number due to three-dimensional effects (see e.g. [45]). The used step size is of $S = 0.49$ and the ratio between the total height H and the inlet height h_{in} is of $H/h_{in} = 1.9423$. We consider here a smaller domain with respect to the experimental setup of Armaly in [45], but sufficient to see the three-dimensional effects. In particular $\frac{x}{S} \in [-10, 20]$, $y \in [-0.49, 0.51]$ and $\frac{z}{S} \in [0, 12]$. The domain is covered using $N_e = 19872$ tetrahedral elements and we take $(p, p_\gamma) = (4, 0)$ and $Re = 600$. We impose the exact Poiseuille profile in the y-direction at the tube inlet, transmissive boundary conditions

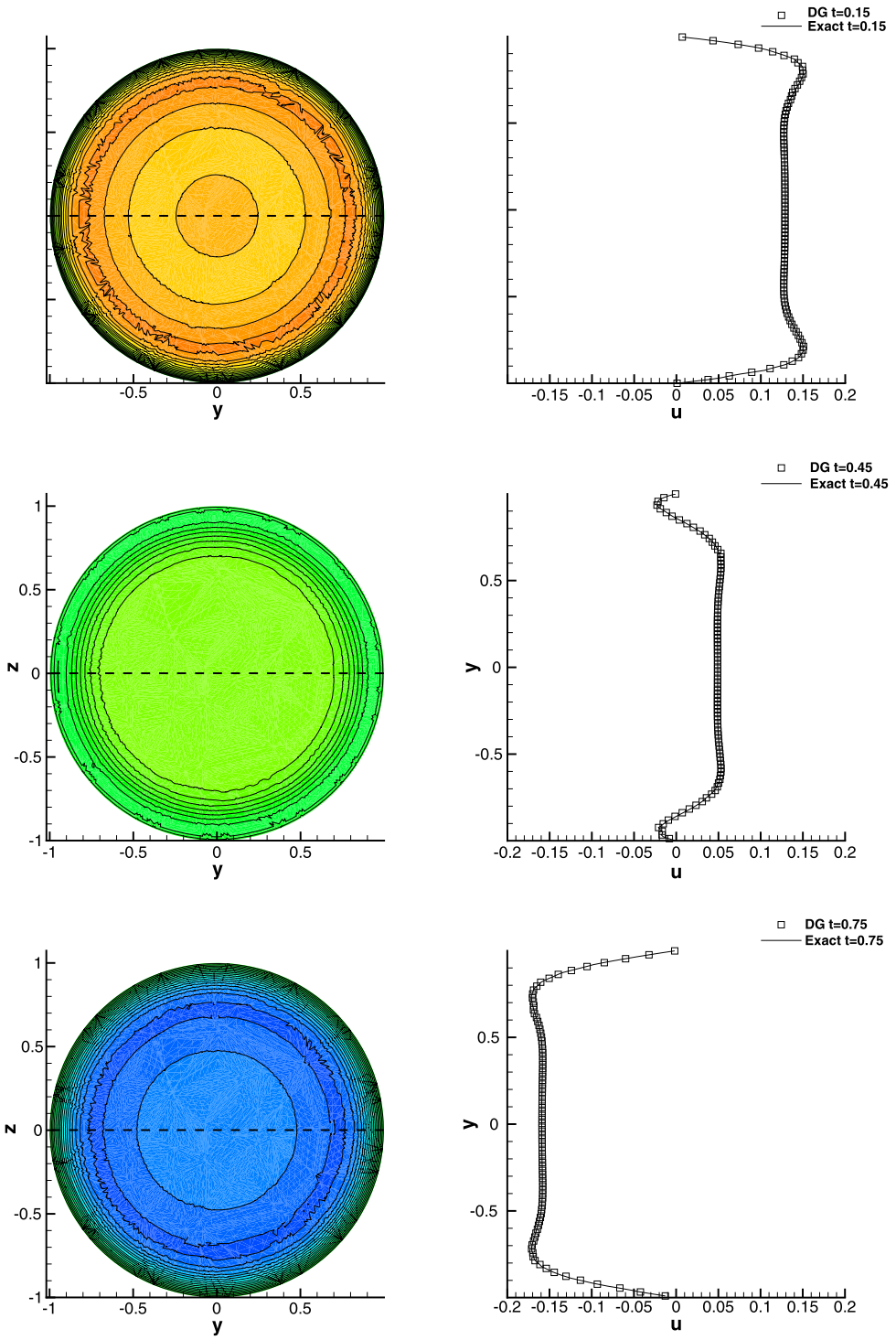

Fig. 11. 3D Womersley flow. Axial velocity contours in the plane $x = 2$ (left column) and comparison of the velocity against the exact solution at $x = 2$ and $z = 0$ (right column) at times, from top to bottom, $t = [0.15, 0.45, 0.75]$.

at the tube outlet and no-slip boundary conditions otherwise. For the current test Δt is taken according to the CFL time restriction based on the magnitude of the flow velocity and $t_{end} = 80$.

A plot of the velocity profile at several values of x/S is shown in Fig. 15. The resulting recirculation zones in the symmetry plane and close to the side wall $\frac{z_{max}}{S}$ are shown in Fig. 16, as well as the equivalent in the plane $(\frac{x}{S}, \frac{z}{S})$ close to the bottom and the top wall in Fig. 17. As we can see, no important secondary recirculation zones appear in the symmetric

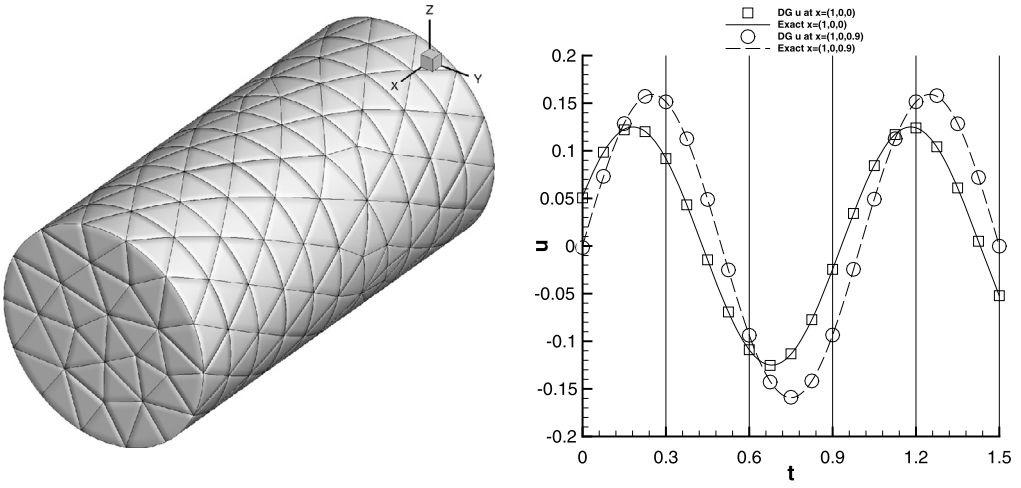

Fig. 12. 3D Womersley flow. Three dimensional view of the isoparametric grid used in this test case (left); Time series of u in the plane $x = 1$, $(y, z) = (0, 0)$ and $(y, z) = (0, 0.9)$ (right). The vertical lines represent the very large time step size of $\Delta t = 0.3$ used in this simulation.

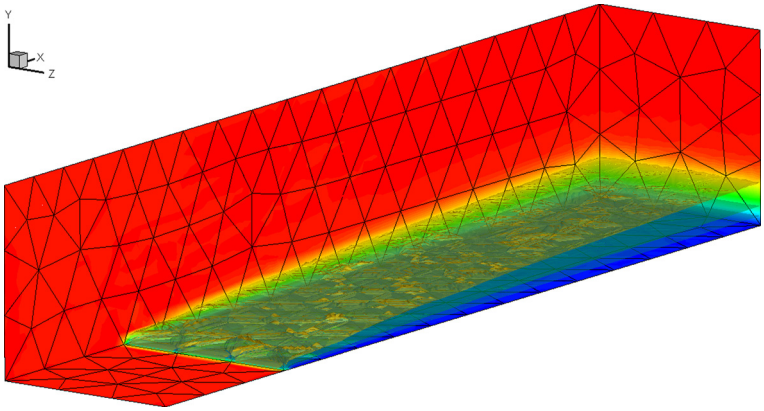

Fig. 13. Blasius boundary layer: $3D$ plot of the domain Ω and sketch of the mesh on the boundary; the plotted iso-surfaces are corresponding to $u = 0.2, 0.4, 0.8$.

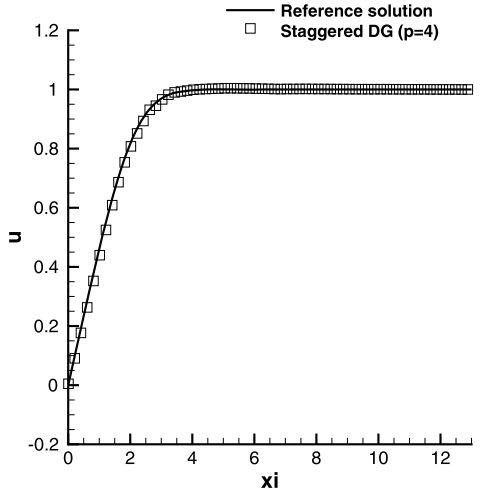

Fig. 14. Blasius boundary layer: numerical solution and reference solution taken on the line $(x, y, z) = (0.4, y, 0)$.

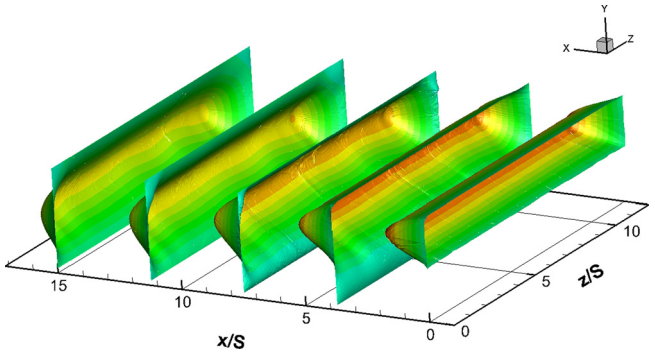

Fig. 15. 3D backward facing step. Value of u in the (y, z)-plane at $x = [0, 3.75, 7.5, 11.25, 15]$.

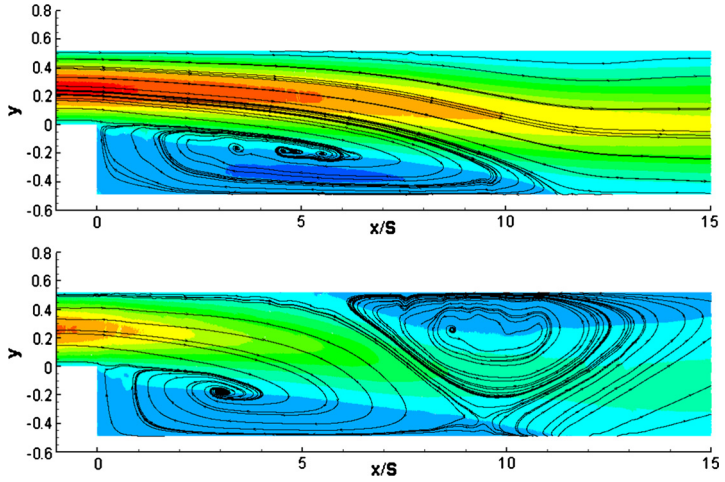

Fig. 16. 3D backward facing step. Recirculation zones in the plane $(\frac{x}{S}, y)$ in the symmetry plane (top) and close to the side wall at $\frac{z}{S} = 12$ (bottom).

plane, while a couple of recirculations appear close to the side walls. The presence of these secondary recirculations seem to reduce the reattachment point for the main recirculation close to the side walls (see Fig. 17 top). On the contrary, a larger recirculation zone can be seen in the middle of the channel. The resulting reattachment point in the symmetry plane is $\frac{x_1}{S} = 11.2$, that is really close to the one obtained in the experimental case, whose value is $\frac{x_1}{S} = 11.24$. Note that the two dimensional numerical simulation, as presented in [61], leads to a reattachment point of $\frac{x_1}{S} = 9.4$, which completely underestimates the experimental one.

3.7. Flow around a sphere

In this section we consider the flow around a sphere. In particular we take as computational domain $\Omega = \mathcal{S}_{10} \cup \mathcal{C}_{10,15} - \mathcal{S}_{0.5}$, where $\mathcal{S}_r$ is a generic sphere with center $\vec{0}$ and radius r; $\mathcal{C}_{r,H}$ is a cylinder with circular basis on the yz-plane, radius r and height H. We use a very coarse grid that is composed by a total number of $N_e = 14403$ tetrahedra whose characteristic length is $h = 0.2$ close to the sphere, while it is only $h = 0.8$ away from the sphere. A sketch of the grid is shown in Fig. 18.

We start from an initial steady flow of magnitude $\mathbf{v}_0 = (u_\infty, 0, 0)$ with $u_\infty = 0.5$ and we impose u_∞ on $\mathcal{S}_{10} \cap \{x \leq 0\}$ as boundary condition; transmissive boundary condition on $\mathcal{C}_{10,15}$ and no-slip condition on $\mathcal{S}_{0.5}$. We use a polynomial degree $(p, p_\gamma) = (3, 0)$ and $\theta = 0.51$ using the method explained in section 2.7; $Re = 300$; $t_{end} = 300$ and Δt is taken according to the CFL time restriction for the convective term.

A plot of the spanwise velocity contours for v is reported in Fig. 19 at $t = 300$ and shows a very complex and three-dimensional behavior of the numerical solution. The mean number of iterations needed to solve the pressure system with a tolerance of $tol = 10^{-8}$ is $I_{mean} = 201.8$ for this test problem. The maximum number of iterations is $I_{max} = 2552$ and is observed only at the beginning of the simulation, when the constant initial condition for the velocity has to be adjusted. Instead, the minimum number of iterations $I_{min} = 62$ is observed when the Von Karman vortex street is fully developed.

A lateral and upper view of a particle tracer is reported in Fig. 20 at $t = 300$, while contour plots of the velocity magnitude are depicted in Fig. 21. The obtained results look very similar to the experimental ones obtained by H. Sakamoto et al. in [44]. The resulting Strouhal number for this simulation is $St = 0.145$, which is close to the experimental range $St = 0.15$–0.18 obtained in [44].

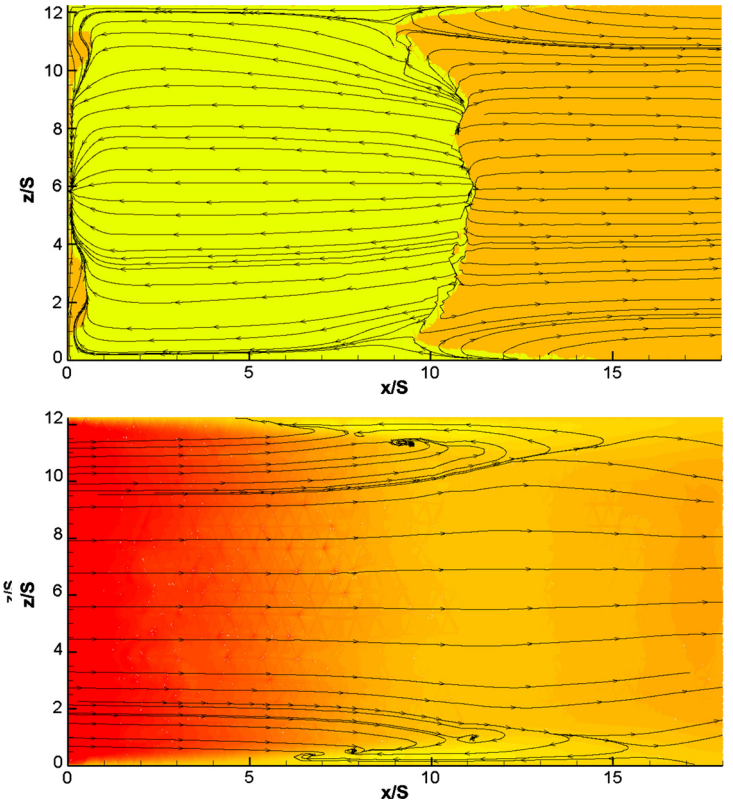

Fig. 17. 3D backward facing step. Recirculation zones in the plane $(\frac{x}{S}, \frac{z}{S})$ close to the bottom and close to the top wall.

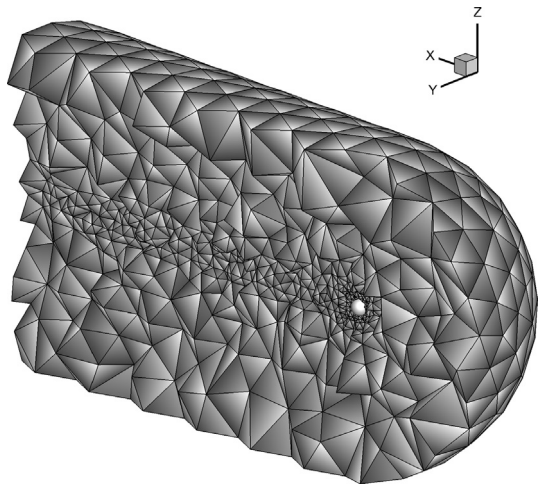

Fig. 18. Flow around a sphere. Cut view of the computational domain with $N_e = 14403$.

3.8. 3D flow past a circular cylinder

In this last test case we want to treat another classical problem for the incompressible Navier–Stokes equations that is the 3D flow around a circular cylinder. For this test, some numerical and experimental cases are available for a large range of Reynolds numbers. In particular several papers focus the attention on the formation of two instability modes characterized by large and small-scale streamwise vortex structures (see e.g. [43]), which act on the Reynolds–Strouhal number relationship. We consider here the problem of the flow past a circular cylinder in a confined channel and for a Reynolds number large enough to have three-dimensional effects and small-scale streamwise vortex structures. We define the blockage ratio $\beta = d/H$ where d indicates the cylinder diameter and H is the distance separating the two walls. In [42] an experimental investigation for a blockage ratio of $\beta = 1/3$ was presented, producing the $Re - St \cdot Re$ relation up to

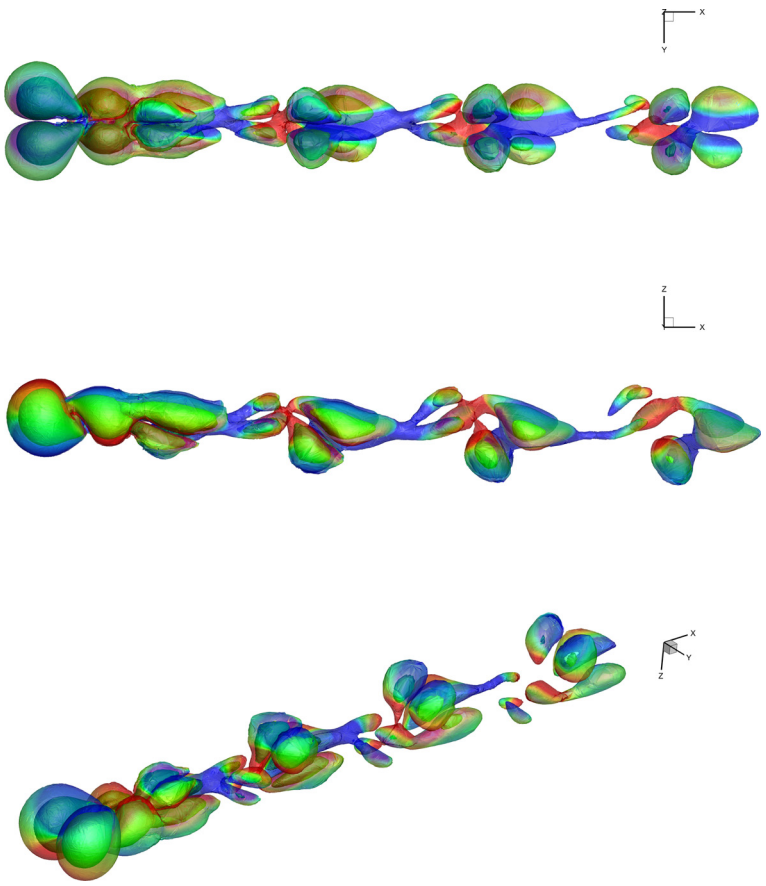

Fig. 19. Flow around a sphere. Contour isosurfaces for the spanwise velocity v in the (x, y) plane, in the (y, z) plane and 3D plot.

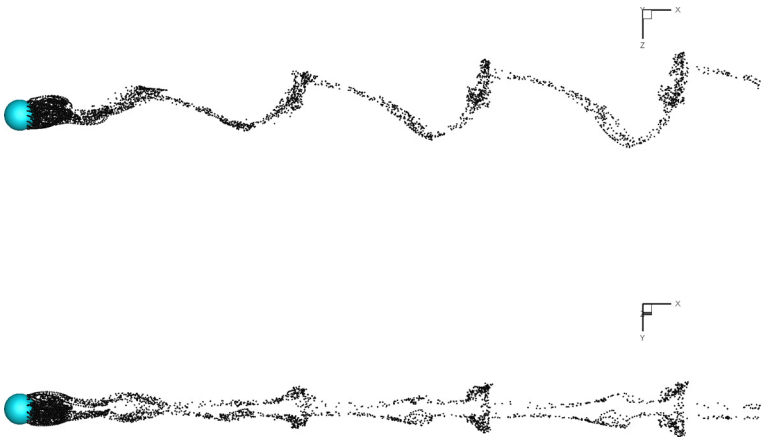

Fig. 20. Flow around a sphere. Side view and upper view of the particle path at $t = 300$.

$Re = 277$. Other numerical studies of Kanaris et al. in [49] give us a numerical analysis in the case of lower blockage ratio of $\beta = 1/5$, finding a similar relation with respect to the unconfined experimental case of Williamson in [43]. We consider here two domains that are $\Omega_1 = [-10, 30] \times [-2.5, 2.5] \times [-12, 12] - C_{0.5,24}$ and $\Omega_2 = [-10, 30] \times [-1.5, 1.5] \times [-12, 12] - C_{0.5,24}$ where $C_{r,z}$ represents the cylinder of radius r and height z centered in 0 and corresponding to a blockage ratio of $\beta = 1/5$ and $\beta = 1/3$, respectively. The first domain Ω_1 is covered with a total number of $N_e = 50761$ tetrahedra and Ω_2 is covered with $N_e = 32527$ elements. A sketch of the grid used in both the cases is shown in Fig. 22. As numerical parameters we use $(p, p_\gamma) = (3, 0)$, $\theta = 0.51$ and $t_{end} = 200$. As initial condition we take a fully developed laminar Poiseuille profile and we impose velocity boundary conditions on the inlet, transmissive boundary conditions on the outlet and no slip boundary conditions otherwise. Finally we impose for the two tests $\nu_1 = 1.66667 \cdot 10^{-3}$ and $\nu_2 = 1.80505 \cdot 10^{-3}$ corresponding to

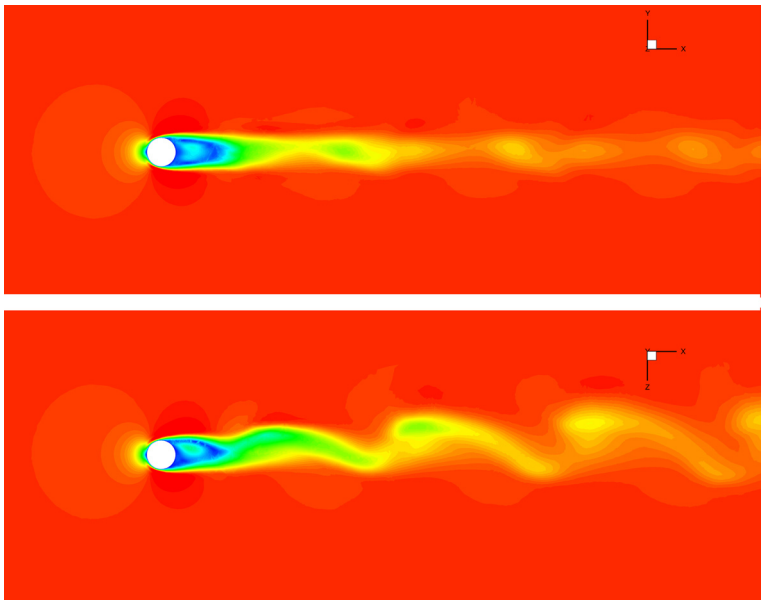

Fig. 21. Flow around a sphere. Velocity magnitude at $t = t_{end}$ in the (x, y) and (x, z)-plane.

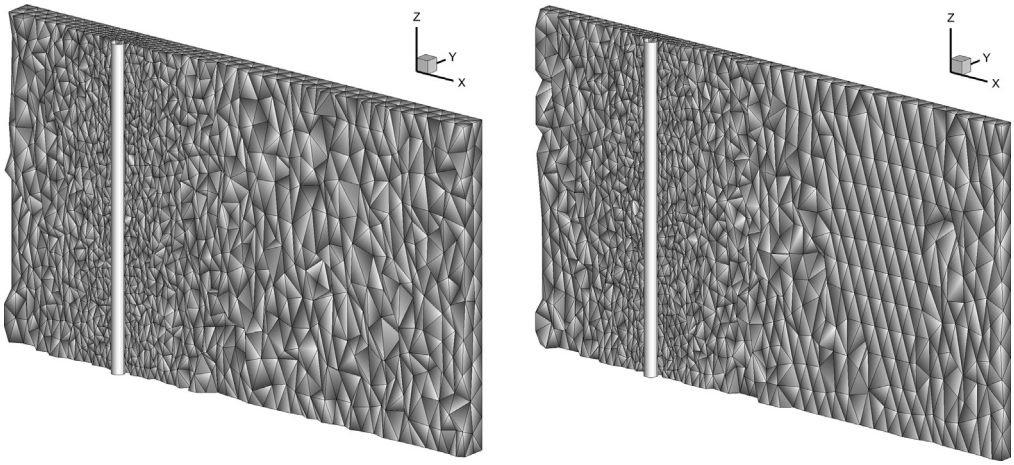

Fig. 22. Flow around a cylinder. Half grid plot of Ω_1 (left) and Ω_2 (right).

$Re_1 = 300$ and $Re_2 = 277$. Furthermore, isoparametric elements are considered for both the cases in order to fit better the curved cylinder. The resulting velocity contours at t_{end} are reported in Fig. 23, where we can observe the generation of the Von Karman vortex street past the cylinder, as well as the three-dimensional mixing effects given by the spanwise velocity w.

The resulting Strouhal number for the first case is $St = 0.198$ which is in good agreement with the numerical one $St = 0.1989$ of Kanaris in [49] and the experimental one of Williamson in [43]. In the second case the obtained Strouhal number is $St = 0.2414$, which corresponds to a value of $St \cdot Re = 66.877$ that is in line with the experimental one of Rehimi et al. in [42], whose extrapolated value is $St \cdot Re = 66.929$. This confirms the suggestion given in [42] that the Strouhal number increases with increasing blockage.

4. Conclusions

In this paper we have proposed a new arbitrary high order accurate space–time DG method on *staggered* unstructured tetrahedral meshes for the solution of the incompressible Navier–Stokes equations in three space dimensions. The key idea of our approach is indeed the use of a *staggered mesh*, where the pressure is defined on the main tetrahedral grid, while the velocity is defined on a face-based staggered dual mesh, composed of non-standard five-point hexahedral elements. To avoid the solution of nonlinear systems due to the presence of the nonlinear convective terms, we opt for a semi-implicit discretization in combination with an outer Picard iteration, leading to a rather simple space–time pressure correction

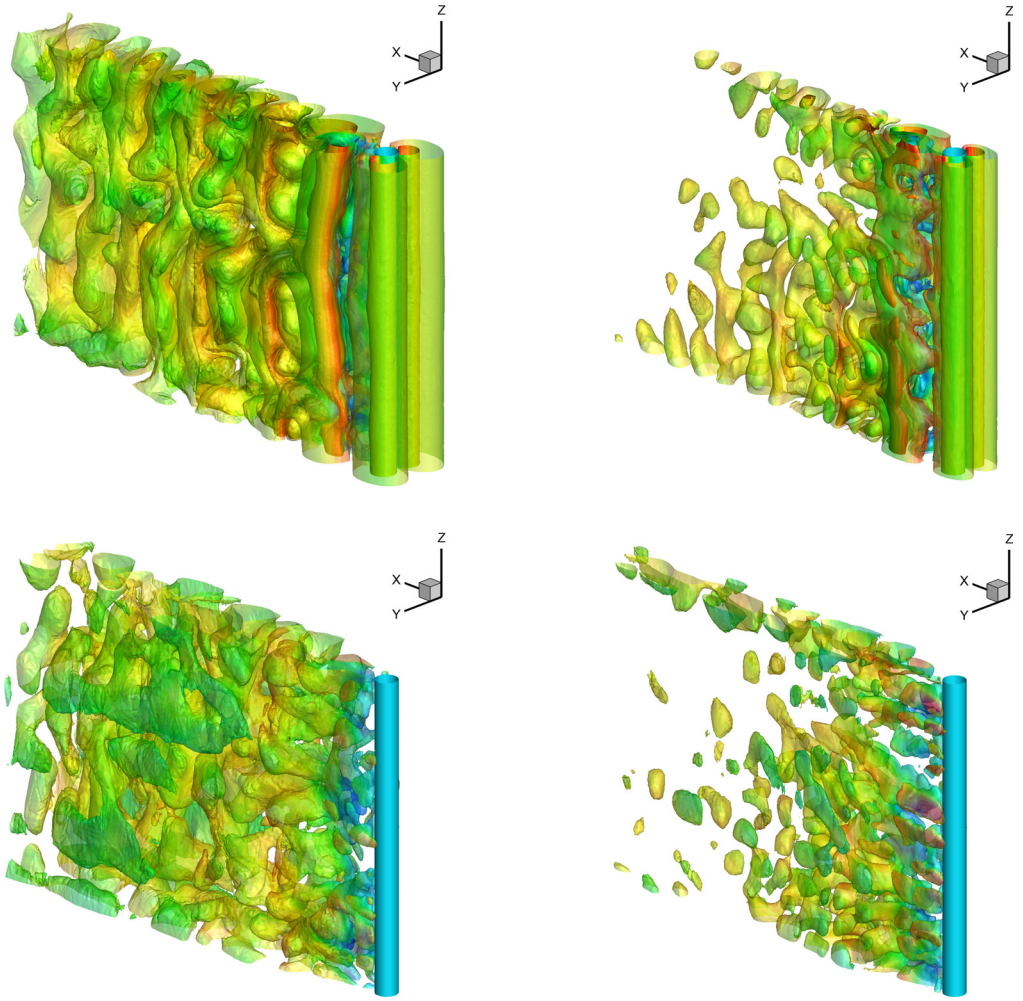

Fig. 23. Flow around a cylinder. Isosurfaces of spanwise velocity $v = [\pm 0.1, \pm 0.03]$ and $w = \pm 0.03$ from top to bottom for the case $(Re, \beta) = (300, \frac{1}{5})$ (left column) and $(Re, \beta) = (277, \frac{1}{3})$ (right column).

algorithm. To the knowledge of the authors, this is the first time that a *staggered* space–time DG scheme has been proposed for the 3D incompressible Navier–Stokes equations on unstructured tetrahedral meshes.

The use of a staggered grid follows the ideas of classical finite difference schemes for the incompressible Navier–Stokes equations, but it is not yet very widespread in the DG community. However, it allows to produce a linear system to be solved in each time step with the smallest number of unknowns (only the scalar pressure) and with the smallest possible stencil (5-point stencil). The same DG algorithm on a *collocated mesh* would either lead to a 17-point stencil (if the pressure is used as the only unknown, substituting the momentum equation into the continuity equation), or to a four times larger linear system with pressure and velocity as unknowns (if a 5-point stencil is used, hence *not* substituting the momentum equation into the continuity equation). In the special case of piecewise constant polynomials in time ($p_\gamma = 0$), the final system matrix becomes even symmetric and positive definite for appropriate boundary conditions, thus allowing the use of the conjugate gradient method. In all test cases shown in this paper, the pressure system could be solved with a simple matrix-free version of the GMRES/CG method, without the use of any preconditioner. In addition, all the coefficient matrices needed by the scheme can be precomputed and stored in a preprocessing step. In this way also the extension to high order isoparametric geometry becomes natural and does not affect the computational effort during run time. The staggered DG approach further allows to avoid the use of numerical flux functions (Riemann solvers) in the scheme, since all quantities are readily defined where they are needed, apart from the nonlinear convective terms, which are treated in a classical manner.

The new numerical method has been applied to a large set of different steady and unsteady benchmark problems. It has been shown that the method achieves high order of accuracy in both, space and time, allowing thus the use of very coarse meshes in space and the use of very large time steps, without compromising the overall accuracy of the method.

Future work will include the extension of the proposed staggered space–time DG method to the compressible Navier–Stokes equations in order to produce a novel family of all Mach number flow solvers, similar to the ideas proposed in [110–118] in the context of semi-implicit finite difference and finite volume schemes for compressible flows.

Acknowledgements

The research presented in this paper was funded by the European Research Council (ERC) under the European Union's Seventh Framework Programme (FP7/2007–2013) within the research project *STiMulUs*, ERC Grant agreement no. 278267.

The authors would also like to acknowledge PRACE for awarding access to the SuperMUC supercomputer based in Munich, Germany, at the Leibniz Rechenzentrum (LRZ).

References

[1] F. Harlow, J. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface, Phys. Fluids 8 (1965) 2182–2189.
[2] V. Patankar, B. Spalding, A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows, Int. J. Heat Mass Transf. 15 (1972) 1787–1806.
[3] V. Patankar, Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Corporation, 1980.
[4] J. van Kan, A second-order accurate pressure correction method for viscous incompressible flow, SIAM J. Sci. Stat. Comput. 7 (1986) 870–891.
[5] C. Taylor, P. Hood, A numerical solution of the Navier–Stokes equations using the finite element technique, Comput. Fluids 1 (1973) 73–100.
[6] A. Brooks, T. Hughes, Stream-line upwind/Petrov Galerkin formulation for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equation, Comput. Methods Appl. Mech. Eng. 32 (1982) 199–259.
[7] T. Hughes, M. Mallet, M. Mizukami, A new finite element formulation for computational fluid dynamics: II. Beyond SUPG, Comput. Methods Appl. Mech. Eng. 54 (1986) 341–355.
[8] M. Fortin, Old and new finite elements for incompressible flows, Int. J. Numer. Methods Fluids 1 (1981) 347–364.
[9] R. Verfürth, Finite element approximation of incompressible Navier–Stokes equations with slip boundary condition II, Numer. Math. 59 (1991) 615–636.
[10] J.G. Heywood, R. Rannacher, Finite element approximation of the nonstationary Navier–Stokes problem. I. Regularity of solutions and second order error estimates for spatial discretization, SIAM J. Numer. Anal. 19 (1982) 275–311.
[11] J.G. Heywood, R. Rannacher, Finite element approximation of the nonstationary Navier–Stokes problem. III. Smoothing property and higher order error estimates for spatial discretization, SIAM J. Numer. Anal. 25 (1988) 489–512.
[12] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, J. Comput. Phys. 131 (1997) 267–279.
[13] C. Baumann, J. Oden, A discontinuous hp finite element method for convection–diffusion problems, Comput. Methods Appl. Mech. Eng. 175 (1999) 311–341.
[14] C. Baumann, J. Oden, A discontinuous hp finite element method for the Euler and Navier–Stokes equations, Int. J. Numer. Methods Fluids 31 (1999) 79–95.
[15] F. Bassi, A. Crivellini, D.D. Pietro, S. Rebay, An artificial compressibility flux for the discontinuous Galerkin solution of the incompressible Navier–Stokes equations, J. Comput. Phys. 218 (2006) 208–221.
[16] F. Bassi, A. Crivellini, D.D. Pietro, S. Rebay, An implicit high-order discontinuous Galerkin method for steady and unsteady incompressible flows, Comput. Fluids 36 (2007) 1529–1546.
[17] G. Gassner, F. Lörcher, C.D. Munz, A contribution to the construction of diffusion fluxes for finite volume and discontinuous Galerkin schemes, J. Comput. Phys. 224 (2007) 1049–1063.
[18] G. Gassner, F. Lörcher, C.D. Munz, A discontinuous Galerkin scheme based on a space–time expansion II. Viscous flow equations in multi dimensions, J. Sci. Comput. 34 (2008) 260–286.
[19] M. Dumbser, Arbitrary high order PNPM schemes on unstructured meshes for the compressible Navier–Stokes equations, Comput. Fluids 39 (2010) 60–76.
[20] R. Hartmann, P. Houston, Symmetric interior penalty DG methods for the compressible Navier–Stokes equations I: method formulation, Int. J. Numer. Anal. Model. 3 (2006) 1–20.
[21] R. Hartmann, P. Houston, An optimal order interior penalty discontinuous Galerkin discretization of the compressible Navier–Stokes equations, J. Comput. Phys. 227 (2008) 9670–9685.
[22] K. Shahbazi, P.F. Fischer, C.R. Ethier, A high-order discontinuous Galerkin method for the unsteady incompressible Navier–Stokes equations, J. Comput. Phys. 222 (2007) 391–407.
[23] H. Luo, L. Luo, R. Nourgaliev, V. Mousseau, N. Dinh, A reconstructed discontinuous Galerkin method for the compressible Navier–Stokes equations on arbitrary grids, J. Comput. Phys. 229 (2010) 6961–6978.
[24] E. Ferrer, R. Willden, A high order discontinuous Galerkin finite element solver for the incompressible Navier–Stokes equations, Comput. Fluids 46 (2011) 224–230.
[25] N. Nguyen, J. Peraire, B. Cockburn, An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations, J. Comput. Phys. 230 (2011) 1147–1170.
[26] A. Crivellini, V. D'Alessandro, F. Bassi, High-order discontinuous Galerkin solutions of three-dimensional incompressible RANS equations, Comput. Fluids 81 (2013) 122–133.
[27] B. Klein, F. Kummer, M. Oberlack, A SIMPLE based discontinuous Galerkin solver for steady incompressible flows, J. Comput. Phys. 237 (2013) 235–250.
[28] B. Cockburn, S.Y. Lin, C.W. Shu, TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one dimensional systems, J. Comput. Phys. 84 (1989) 90–113.
[29] B. Cockburn, S. Hou, C.W. Shu, The Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case, Math. Comput. 54 (1990) 545–581.
[30] B. Cockburn, C.W. Shu, The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems, J. Comput. Phys. 141 (1998) 199–224.
[31] J.J.W. van der Vegt, H. van der Ven, Space—time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows I. General formulation, J. Comput. Phys. 182 (2002) 546–585.
[32] H. van der Ven, J.J.W. van der Vegt, Space–time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows II. Efficient flux quadrature, Comput. Methods Appl. Mech. Eng. 191 (2002) 4747–4780.

[33] C. Klaij, J.V. der Vegt, H.V. der Ven, Space–time discontinuous Galerkin method for the compressible Navier–Stokes equations, J. Comput. Phys. 217 (2006) 589–611.
[34] S. Rhebergen, B. Cockburn, A space–time hybridizable discontinuous Galerkin method for incompressible flows on deforming domains, J. Comput. Phys. 231 (2012) 4185–4204.
[35] S. Rhebergen, B. Cockburn, J.J. van der Vegt, A space–time discontinuous Galerkin method for the incompressible Navier–Stokes equations, J. Comput. Phys. 233 (2013) 339–358.
[36] M. Balazsova, M. Feistauer, On the stability of the ALE space–time discontinuous Galerkin method for nonlinear convection–diffusion problems in time-dependent domains, Appl. Math. 60 (2015) 501–526.
[37] M. Balazsova, M. Feistauer, M. Hadrava, A. Kosik, On the stability of the space–time discontinuous Galerkin method for the numerical solution of nonstationary nonlinear convection–diffusion problems, J. Numer. Math. 23 (2015) 211–233.
[38] M. Dumbser, M. Facchini, A local space–time discontinuous Galerkin method for Boussinesq-type equations, Appl. Math. Comput. 272 (2016) 336–346.
[39] M. Tavelli, M. Dumbser, A staggered arbitrary high order semi-implicit discontinuous Galerkin method for the two dimensional incompressible Navier–Stokes equations, Comput. Fluids 119 (2015) 235–249.
[40] L. Wang, P. Persson, A high-order discontinuous Galerkin method with unstructured space–time meshes for two-dimensional compressible flows on domains with large deformations, Comput. Fluids 118 (2015) 53–68.
[41] L. Demkowicz, J. Oden, T. Strouboulis, Adaptive finite elements for flow problems with moving boundaries. Part I: variational principles and a posteriori estimates, Comput. Methods Appl. Mech. Eng. 46 (1984) 217–251.
[42] F. Remihi, F. Aloui, S.B. Nasrallah, L. Doubliez, J. Legrand, Experimental investigation of a confined flow downstream of a circular cylinder centred between two parallel walls, J. Fluids Struct. 24 (2008) 855–882.
[43] C. Williamson, The existence of two stages in the transition to three-dimensionality of a cylinder wake, Phys. Fluids 24 (1988) 855–882.
[44] H. Sakamoto, H. Haniu, A study on vortex shedding from spheres in a uniform flow, J. Fluids Eng. 112 (1990) 386–392.
[45] B. Armaly, F. Durst, J. Pereira, B. Schonung, Experimental and theoretical investigation on backward-facing step flow, J. Fluid Mech. 127 (1983) 473–496.
[46] H.C. Ku, R.S. Hirsh, T.D. Taylor, A pseudospectral method for solution of the three-dimensional incompressible Navier–Stokes equations, J. Comput. Phys. 70 (1987) 439–462.
[47] C.K. Aidun, N. Triantafillopoulos, J. Benson, Global stability of a lid-driven cavity with throughflow: flow visualization studies, Phys. Fluids 3 (1991) 2081–2091.
[48] M. Brachet, D. Meiron, S. Orszag, Small-scale structure of the Taylor–Green vortex, J. Fluid Mech. 130 (1983) 411–452.
[49] N. Kanaris, D. Grigoriadis, S. Kassinos, Three dimensional flow around a circular cylinder confined in a plane channel, Phys. Fluids 23 (2011) 064106.
[50] V. Ribeiro, P. Coelho, F. Pinho, M. Alves, Three-dimensional effects in laminar flow past a confined cylinder, Chem. Eng. Sci. 84 (2012) 155–169.
[51] A. Kravchenko, P. Moin, Numerical studies of flow over a circular cylinder at $Re_D = 3900$, Phys. Fluids 12 (2000) 403.
[52] M.R. Hestenes, E. Stiefel, Methods of conjugate gradients for solving linear systems, J. Res. Natl. Bur. Stand. 49 (1952) 409–436.
[53] Y. Saad, M. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (1986) 856–869.
[54] D.N. Arnold, F. Brezzi, B. Cockburn, L.D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM J. Numer. Anal. 39 (2002) 1749–1779.
[55] Y.J. Liu, C.W. Shu, E. Tadmor, M. Zhang, Central discontinuous Galerkin methods on overlapping cells with a non-oscillatory hierarchical reconstruction, SIAM J. Numer. Anal. 45 (2007) 2442–2467.
[56] Y.J. Liu, C.W. Shu, E. Tadmor, M. Zhang, L2-stability analysis of the central discontinuous Galerkin method and a comparison between the central and regular discontinuous Galerkin methods, Math. Model. Numer. Anal. 42 (2008) 593–607.
[57] E. Chung, B. Engquist, Optimal discontinuous Galerkin methods for wave propagation, SIAM J. Numer. Anal. 44 (2006) 2131–2158.
[58] E.T. Chung, C.S. Lee, A staggered discontinuous Galerkin method for the convection–diffusion equation, J. Numer. Math. 20 (2012) 1–31.
[59] M. Dumbser, V. Casulli, A staggered semi-implicit spectral discontinuous Galerkin scheme for the shallow water equations, Appl. Math. Comput. 219 (15) (2013) 8057–8077.
[60] M. Tavelli, M. Dumbser, A high order semi-implicit discontinuous Galerkin method for the two dimensional shallow water equations on staggered unstructured meshes, Appl. Math. Comput. 234 (2014) 623–644.
[61] M. Tavelli, M. Dumbser, A staggered semi-implicit discontinuous Galerkin method for the two dimensional incompressible Navier–Stokes equations, Appl. Math. Comput. 248 (2014) 70–92.
[62] S. Cheung, E. Chung, H. Kim, Y. Qian, Staggered discontinuous Galerkin methods for the incompressible Navier–Stokes equations, J. Comput. Phys. 302 (2015) 251–266.
[63] V. Dolejsi, Semi-implicit interior penalty discontinuous Galerkin methods for viscous compressible flows, Commun. Comput. Phys. 4 (2008) 231–274.
[64] V. Dolejsi, M. Feistauer, A semi-implicit discontinuous Galerkin finite element method for the numerical solution of inviscid compressible flow, J. Comput. Phys. 198 (2004) 727–746.
[65] V. Dolejsi, M. Feistauer, J. Hozman, Analysis of semi-implicit DGFEM for nonlinear convection–diffusion problems on nonconforming meshes, Comput. Methods Appl. Mech. Eng. 196 (2007) 2813–2827.
[66] F.X. Giraldo, M. Restelli, High-order semi-implicit time-integrators for a triangular discontinuous Galerkin oceanic shallow water model, Int. J. Numer. Methods Fluids 63 (2010) 1077–1102.
[67] G. Tumolo, L. Bonaventura, M. Restelli, A semi-implicit, semi-Lagrangian, p-adaptive discontinuous Galerkin method for the shallow water equations, J. Comput. Phys. 232 (2013) 46–67.
[68] C.W. Hirt, B.D. Nichols, Volume of fluid (VOF) method for dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201–225.
[69] V. Casulli, R.T. Cheng, Semi-implicit finite difference methods for three-dimensional shallow water flow, Int. J. Numer. Methods Fluids 15 (1992) 629–648.
[70] V. Casulli, A semi-implicit finite difference method for non-hydrostatic free-surface flows, Int. J. Numer. Methods Fluids 30 (1999) 425–440.
[71] V. Casulli, R.A. Walters, An unstructured grid, three-dimensional model based on the shallow water equations, Int. J. Numer. Methods Fluids 32 (2000) 331–348.
[72] V. Casulli, A high-resolution wetting and drying algorithm for free-surface hydrodynamics, Int. J. Numer. Methods Fluids 60 (2009) 391–408.
[73] V. Casulli, A semi-implicit numerical method for the free-surface Navier–Stokes equations, Int. J. Numer. Methods Fluids 74 (2014) 605–622.
[74] L. Brugnano, V. Casulli, Iterative solution of piecewise linear systems, SIAM J. Sci. Comput. 30 (2007) 463–472.
[75] L. Brugnano, V. Casulli, Iterative solution of piecewise linear systems and applications to flows in porous media, SIAM J. Sci. Comput. 31 (2009) 1858–1873.
[76] V. Casulli, P. Zanolli, A nested Newton–type algorithm for finite volume methods solving Richards' equation in mixed form, SIAM J. Sci. Comput. 32 (2009) 2255–2273.
[77] V. Casulli, P. Zanolli, Iterative solutions of mildly nonlinear systems, J. Comput. Appl. Math. 236 (2012) 3937–3947.
[78] A. Bermudez, A. Dervieux, J. Desideri, M. Vazquez, Upwind schemes for the two-dimensional shallow water equations with variable depth using unstructured meshes, Comput. Methods Appl. Mech. Eng. 155 (1998) 49–72.

[79] A. Bermúdez, J. Ferrín, L. Saavedra, M. Vázquez-Cendón, A projection hybrid finite volume/element method for low-Mach number flows, J. Comput. Phys. 271 (2014) 360–378.
[80] E.F. Toro, A. Hidalgo, M. Dumbser, FORCE schemes on unstructured meshes I: conservative hyperbolic systems, J. Comput. Phys. 228 (2009) 3368–3389.
[81] B. Cockburn, C.W. Shu, The local discontinuous Galerkin method for time-dependent convection–diffusion systems, SIAM J. Numer. Anal. 35 (1998) 2440–2463.
[82] B. Cockburn, C.W. Shu, Runge–Kutta discontinuous Galerkin methods for convection–dominated problems, J. Sci. Comput. 16 (2001) 173–261.
[83] J. Yan, C. Shu, A local discontinuous Galerkin method for KdV-type equations, SIAM J. Numer. Anal. 40 (2002) 769–791.
[84] V.V. Rusanov, Calculation of interaction of non-steady Shock waves with obstacles, J. Comput. Math. Phys. USSR 1 (1961) 267–279.
[85] V. Chatzi, F. Preparata, Using Pyramids in Mixed Meshes – Point Placement and Basis Functions, Brown University, Providence, 2000.
[86] H. Luo, J. Baum, R. Löhner, A discontinuous Galerkin method based on a Taylor basis for the compressible flows on arbitrary grids, J. Comput. Phys. 227 (2008) 8875–8893.
[87] M.J. Castro, J.M. Gallardo, C. Parés, High-order finite volume schemes based on reconstruction of states for solving hyperbolic systems with nonconservative products. applications to shallow-water systems, Math. Comput. 75 (2006) 1103–1134.
[88] C. Parés, Numerical methods for nonconservative hyperbolic systems: a theoretical framework, SIAM J. Numer. Anal. 44 (2006) 300–321.
[89] D.L. Brown, R. Cortez, M.L. Minion, Accurate projection methods for the incompressible Navier–Stokes equations, J. Comput. Phys. 168 (2001) 464–499.
[90] M. Dumbser, D.S. Balsara, E.F. Toro, C.D. Munz, A unified framework for the construction of one-step finite-volume and discontinuous Galerkin schemes, J. Comput. Phys. 227 (2008) 8209–8253.
[91] D. Balsara, C. Meyer, M. Dumbser, H. Du, Z. Xu, Efficient implementation of ADER schemes for Euler and magnetohydrodynamical flows on structured meshes – speed comparisons with Runge–Kutta methods, J. Comput. Phys. 235 (2013) 934–969.
[92] A. Dutt, L. Greengard, V. Rokhlin, Spectral deferred correction methods for ordinary differential equations, BIT Numer. Math. 40 (2) (2000) 241–266.
[93] S. Kadioglu, R. Klein, M. Minion, A fourth-order auxiliary variable projection method for zero-Mach number gas dynamics, J. Comput. Phys. 227 (3) (2008) 2012–2043.
[94] V. Casulli, E. Cattani, Stability, accuracy and efficiency of a semi-implicit method for three-dimensional shallow water flow, Comput. Math. Appl. 27 (1994) 99–112.
[95] U. Ghia, K.N. Ghia, C.T. Shin, High-re solutions for incompressible flow using Navier–Stokes equations and multigrid method, J. Comput. Phys. 48 (1982) 387–411.
[96] E. Erturk, T. Corke, C. Gokcol, Numerical solutions of 2D steady incompressible driven cavity flow at high Reynolds numbers, Int. J. Numer. Methods Fluids 48 (2005) 747–774.
[97] S. Albensoeder, H. Kuhlmann, Accurate three-dimensional lid-driven cavity flow, J. Comput. Phys. 206 (2005) 536–558.
[98] V. Arnold, Sur la topologie des écoulements stationnaires des fluides parfaits, C. R. Hebd. Séances Acad. Sci. 261 (1965) 17–20.
[99] S. Childress, New solutions of the kinematic dynamo problem, J. Math. Phys. 11 (1970) 3063–3076.
[100] T. Dombre, U. Frisch, J.M. Greene, M. Hénon, A. Mehr, A.M. Soward, Chaotic streamlines in the ABC flows, J. Fluid Mech. 167 (1986) 353–391.
[101] O.M. Podvigina, Spatially-periodic steady solutions to the three-dimensional Navier–Stokes equation with the abc-force, Physica D: Nonlinear Phenomena 128 (24) (1999) 250–272.
[102] O. Podvigina, P. Ashwin, D. Hawker, Modelling instability of flow using a mode interaction between steady and Hopf bifurcations with rotational symmetries of the cube, Physica D: Nonlinear Phenomena 215 (1) (2006) 62–79.
[103] S. Ershkov, Non-stationary helical flows for incompressible 3D Navier–Stokes equations, Appl. Math. Comput. 274 (2015) 611–614.
[104] F. Fambri, M. Dumbser, Spectral semi-implicit space–time discontinuous Galerkin methods for the incompressible Navier–Stokes equations on staggered Cartesian grids, Appl. Numer. Math. submitted for publication, arXiv:1602.05806.
[105] O. Colomes, S. Badia, R. Codina, J. Principe, Assessment of variational multiscale models for the large eddy simulation of turbulent incompressible flows, Comput. Methods Appl. Mech. Eng. 285 (2015) 32–63.
[106] J. Womersley, Method for the calculation of velocity, rate of flow and viscous drag in arteries when the pressure gradient is known, J. Physiol. 127 (1955) 553–563.
[107] F. Fambri, M. Dumbser, V. Casulli, An efficient semi-implicit method for three-dimensional non-hydrostatic flows in compliant arterial vessels, Int. J. Numer. Methods Biomed. Eng. 30 (2014) 1170–1198.
[108] H. Blasius, Grenzschichten in Flüssigkeiten mit kleiner Reibung, Z. Math. Phys. 56 (1908) 1–37.
[109] E. Erturk, Numerical solutions of 2d steady incompressible flow over a backward-facing step, part I: high Reynolds number solutions, Comput. Fluids 37 (2008) 633–655.
[110] V. Casulli, D. Greenspan, Pressure method for the numerical solution of transient, compressible fluid flows, Int. J. Numer. Methods Fluids 4 (11) (1984) 1001–1012.
[111] R. Klein, N. Botta, T. Schneider, C. Munz, S. Roller, A. Meister, L. Hoffmann, T. Sonar, Asymptotic adaptive methods for multi-scale problems in fluid mechanics, J. Eng. Math. 39 (2001) 261–343.
[112] C. Munz, R. Klein, S. Roller, K. Geratz, The extension of incompressible flow solvers to the weakly compressible regime, Comput. Fluids 32 (2003) 173–196.
[113] R. Klein, Semi-implicit extension of a Godunov-type scheme based on low Mach number asymptotics I: one-dimensional flow, J. Comput. Phys. 121 (1995) 213–237.
[114] A. Meister, Asymptotic single and multiple scale expansions in the low Mach number limit, SIAM J. Appl. Math. 60 (1) (1999) 256–271.
[115] J. Park, C. Munz, Multiple pressure variables methods for fluid flow at all Mach numbers, Int. J. Numer. Methods Fluids 49 (2005) 905–931.
[116] F. Cordier, P. Degond, A. Kumbaro, An asymptotic-preserving all-speed scheme for the Euler and Navier–Stokes equations, J. Comput. Phys. 231 (2012) 5685–5704.
[117] M. Dumbser, U. Iben, M. Ioriatti, An efficient semi-implicit finite volume method for axially symmetric compressible flows in compliant tubes, Appl. Numer. Math. 89 (2015) 24–44.
[118] M. Dumbser, V. Casulli, A conservative, weakly nonlinear semi-implicit finite volume method for the compressible Navier–Stokes equations with general equation of state, Appl. Math. Comput. 272 (2016) 479–497.